Grid Computing and the Globus Toolkit

Raj Kettimuthu

Argonne National Laboratory and

The University of Chicago

IWSGC10

What is a Grid?

- Resource sharing
  - Computers, storage, sensors, networks, …
  - Sharing always conditional: issues of trust, policy, negotiation, payment, …
- Coordinated problem solving
  - Beyond client-server: distributed data analysis, computation, collaboration, …
- Dynamic, multi-institutional virtual organizations
  - Community overlays on classic org structures
  - Large or small, static or dynamic

the globus alliance
www.globus.org

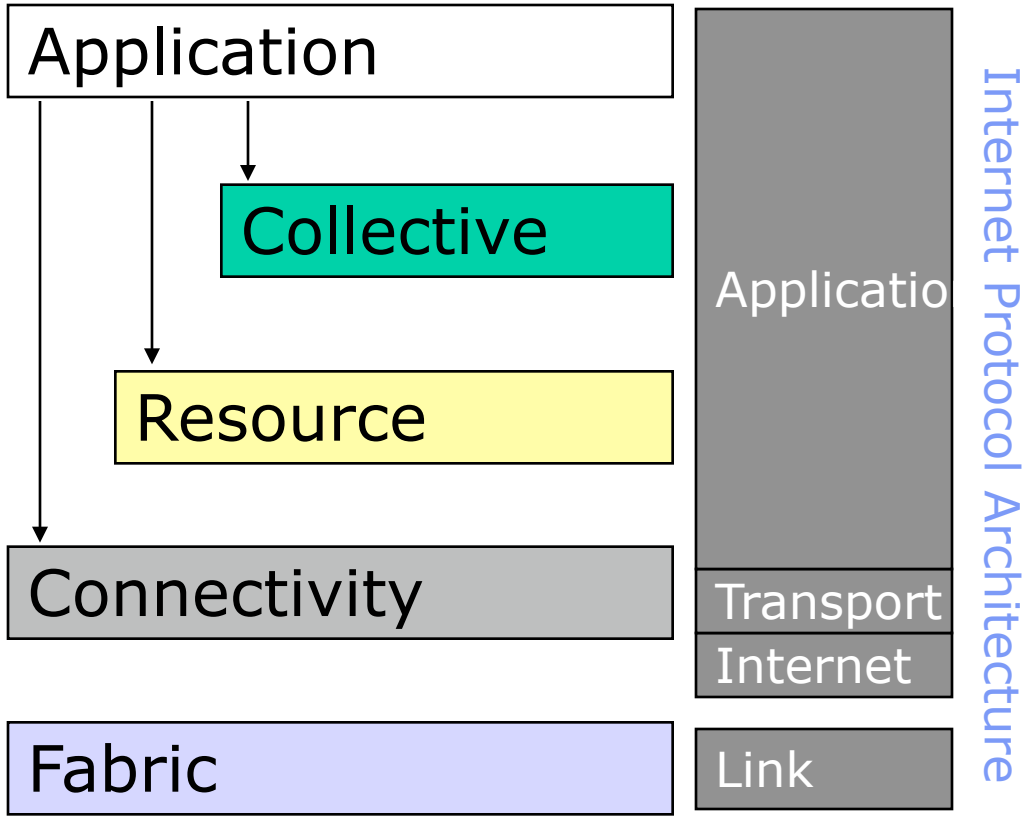
The Grid

"Coordinating multiple resources": ubiquitous infrastructure services, app-specific distributed services

"Sharing single resources": negotiating access, controlling use

"Talking to things": communication (Internet protocols) & security

"Controlling things locally": Access to, & control of, resources



Why Is this Hard or Different?

- Lack of central control
  - Where things run
  - When they run
  - Who can run
- Shared resources
  - Contention, variability
- Communication and coordination
  - Different sites implies different sys admins, users, institutional goals, and often socio-political constraints

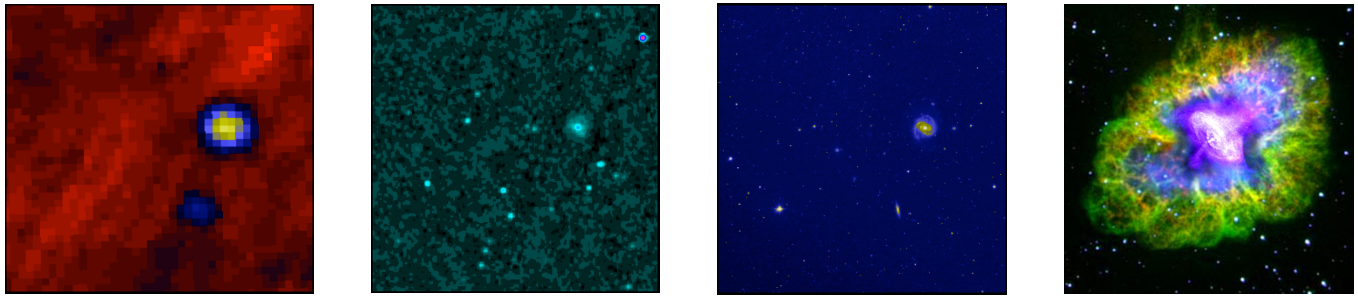
So Why Do It?

- Computations that need to be done with a time limit
- Data that can't fit on one site
- Data owned by multiple sites
- Applications that need to be run bigger, faster, more

the globus alliance
www.globus.org

For Example: Digital Astronomy

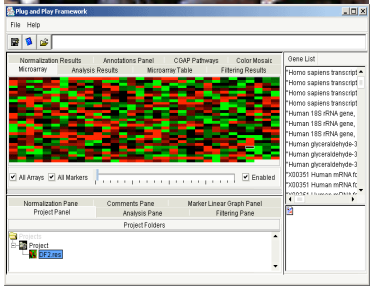
- Digital observatories provide online archives of data at different wavelengths

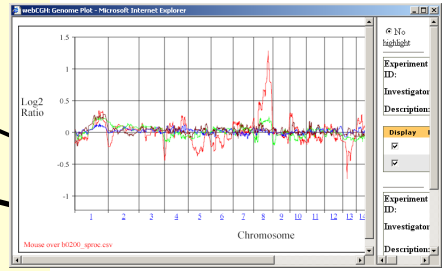


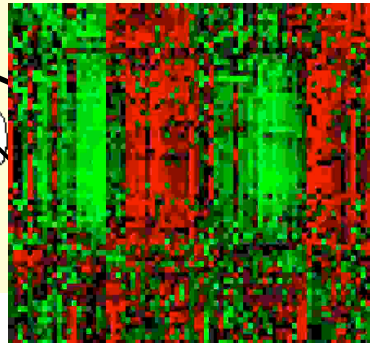
- Ask questions such as: what objects are visible in infrared but not visible spectrum?

the globus alliance
www.globus.org

For Example: Cancer Biology







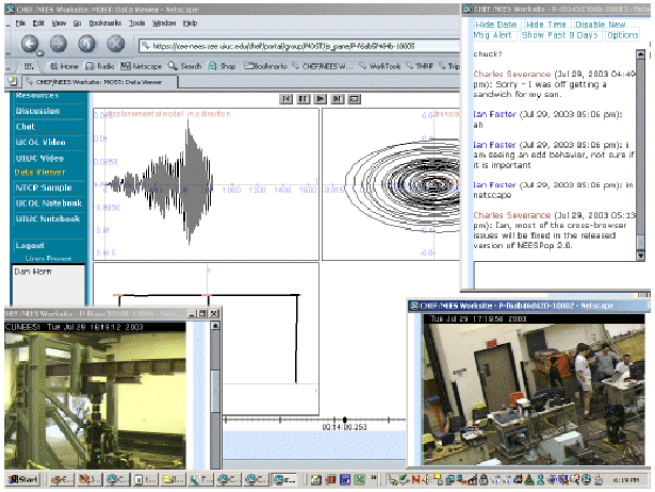
What Kinds of Applications?

- Computation intensive
  - Interactive simulation (climate modeling)
  - Large-scale simulation and analysis (galaxy formation, gravity waves, event simulation)
  - Engineering (parameter studies, linked models)
- Data intensive
  - Experimental data analysis (e.g., physics)
  - Image & sensor analysis (astronomy, climate)
- Distributed collaboration
  - Online instrumentation (microscopes, x-ray)
    Remote visualization (climate studies, biology)
  - Engineering (large-scale structural testing)

Key Common Features

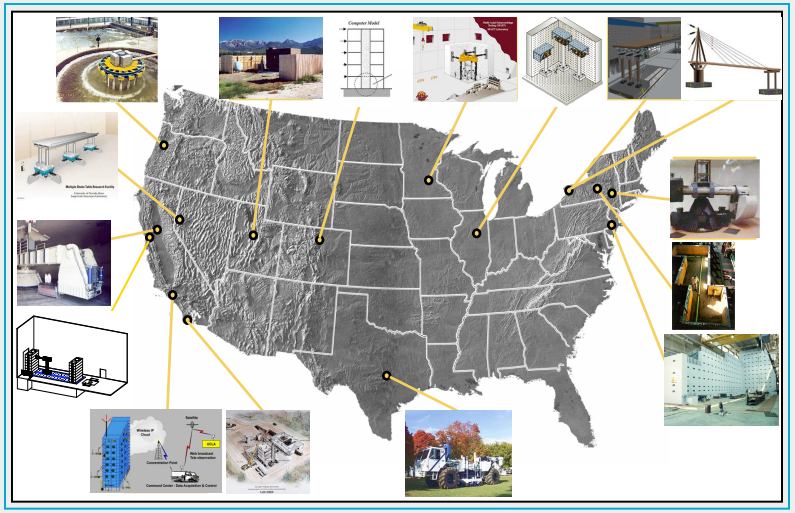
- The size and/or complexity of the problem
- Collaboration between people in several organizations
- Sharing computing resources, data, instruments

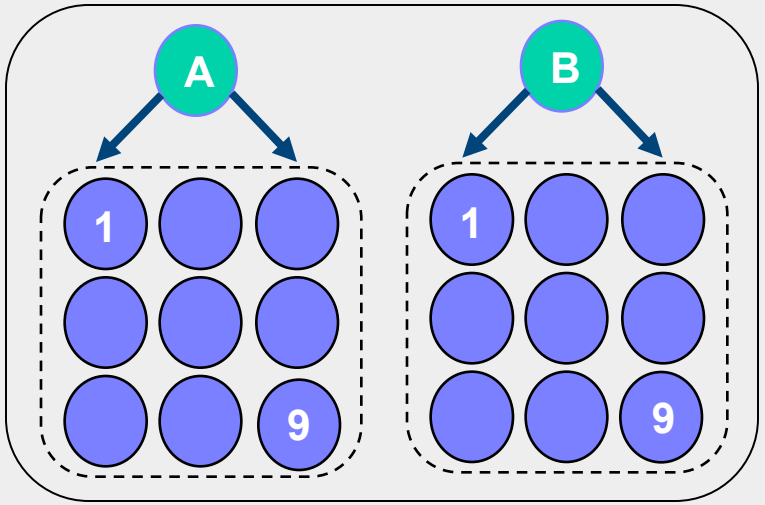
Underlying Problem: The Application-Infrastructure Gap



Dynamic and/or Distributed Applications

Shared Distributed Infrastructure





Grid Infrastructure

- Distributed use and management
  - Of physical resources
  - Of software services
  - Of communities and their policies

the globus alliance
www.globus.org

Globus is…

- A collection of solutions to problems that come up frequently when building collaborative distributed applications
- Software for Grid infrastructure
- Tools to build applications that exploit Grid infrastructure
- Open source & open standards
- Enabler of a rich tool & service ecosystem

Globus Toolkit

The Globus Toolkit centers around

Connectivity layer:

Security: Grid Security Infrastructure (GSI) - allows collaborators to share resources without blind trust

Resource layer:

Resource Management: Grid Resource Allocation Management (GRAM)

Data Transfer: Grid File Transfer Protocol (GridFTP)

Also collective layer protocol

Replica Management (RLS)

- Focuses on simplifying heterogeneity for application developers

Globus is a Building Block

- Basic components for Grid functionality
  - Not turnkey solutions, but building blocks & tools for application developers & system integrators
- Highest-level services are often application specific, we let apps concentrate there
- Easier to reuse than to reinvent
  - Compatibility with other Grid systems comes for free
- We provide basic infrastructure to get you one step closer

Globus Philosophy

- Globus was first established as an open source project in 1996
- The Globus Toolkit is open source to:
  - Allow for inspection
    - for consideration in standardization processes
  - Encourage adoption
    - in pursuit of ubiquity and interoperability
  - Encourage contributions
    - harness the expertise of the community
- The Globus Toolkit is distributed under the (BSD-style) Apache License version 2

Globus Technology Areas

- Security
  - Apply uniform policy across distinct systems
- Execution management
  - Provision, deploy, & manage services
- Data management
  - Discover, transfer, & access large data

Globus User Community

- Large & diverse
  - 10s of national Grids, 100s of applications, 1000s of users; probably much more
  - Every continent except Antarctica
  - Applications ranging across many sciences
  - Dozens (at least) of commercial deployments
- Successful
  - Many production systems doing real work
  - Many applications producing real results
- Smart, energetic, demanding
  - Constant stream of new use cases & tools

Global Community

Examples of Production Scientific Grids

- APAC (Australia)
- China Grid
- China National Grid
- DGrid (Germany)
- EGEE
- NAREGI (Japan)
- Open Science Grid
- Taiwan Grid
- TeraGrid
- ThaiGrid
- UK Nat'l Grid Service

More Specifically, I May Want To …

- Manage who is allowed to access my service or my experimental data or …
- Ensure reliable & secure distribution of data from my lab to my partners
- Run 10,000 jobs on whatever computers I can get hold of

More Specifically, I May Want To …

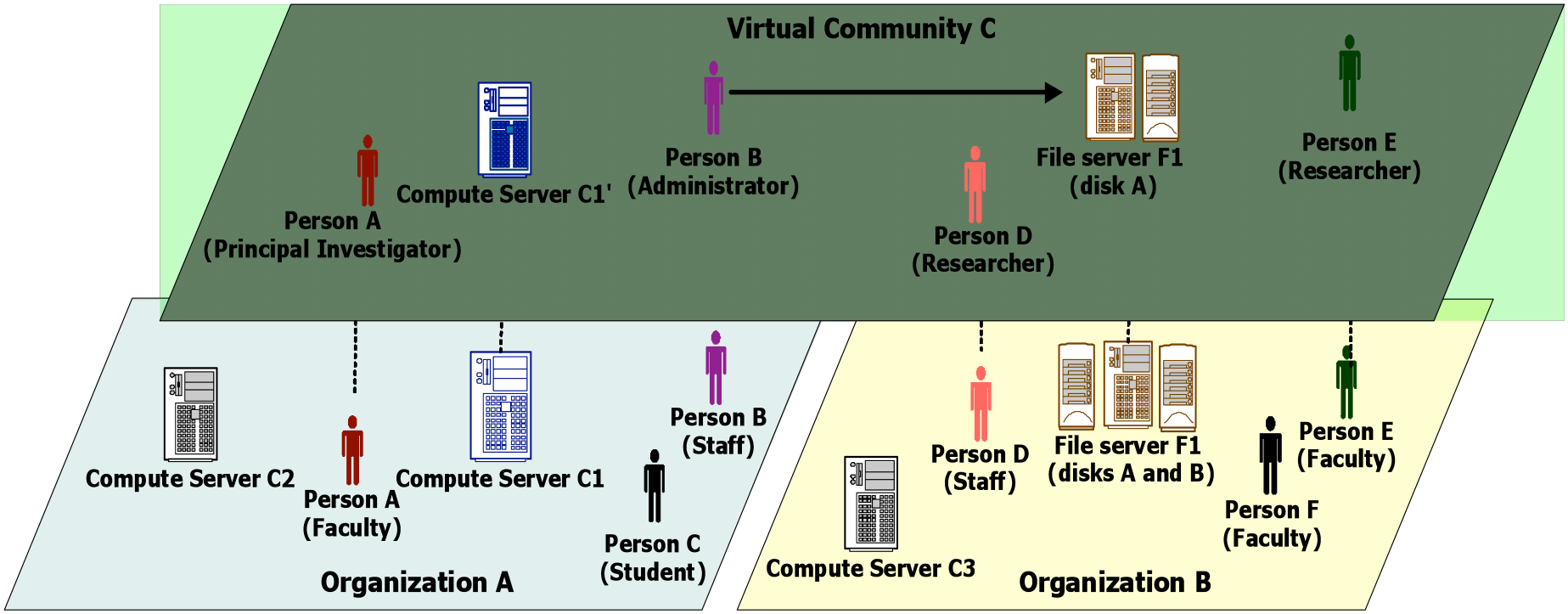
- Manage who is allowed to access my service or my experimental data or …
- Ensure reliable & secure distribution of data from my lab to my partners
- Run 10,000 jobs on whatever computers I can get hold of

Grid Security Concerns

- Control access to shared services
  - Address autonomous management, e.g., different policy in different work groups
- Support multi-user collaborations
  - Federate through mutually trusted services
  - Local policy authorities rule
- Allow users and application communities to set up dynamic trust domains
  - Personal/VO collection of resources working together based on trust of user/VO

the globus alliance

Virtual Organizations (VO)



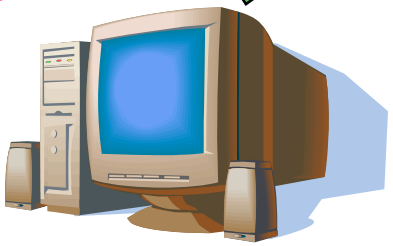
- VO for each application or workload
- Carve out and configure resources for a particular use and set of users

Security - Terminology

Delegation: Granting a right to another entity.

Authorization: What are you allowed to do?

John Doe @ NCSA



Authentication: Proving who you are.

Security - Terminology

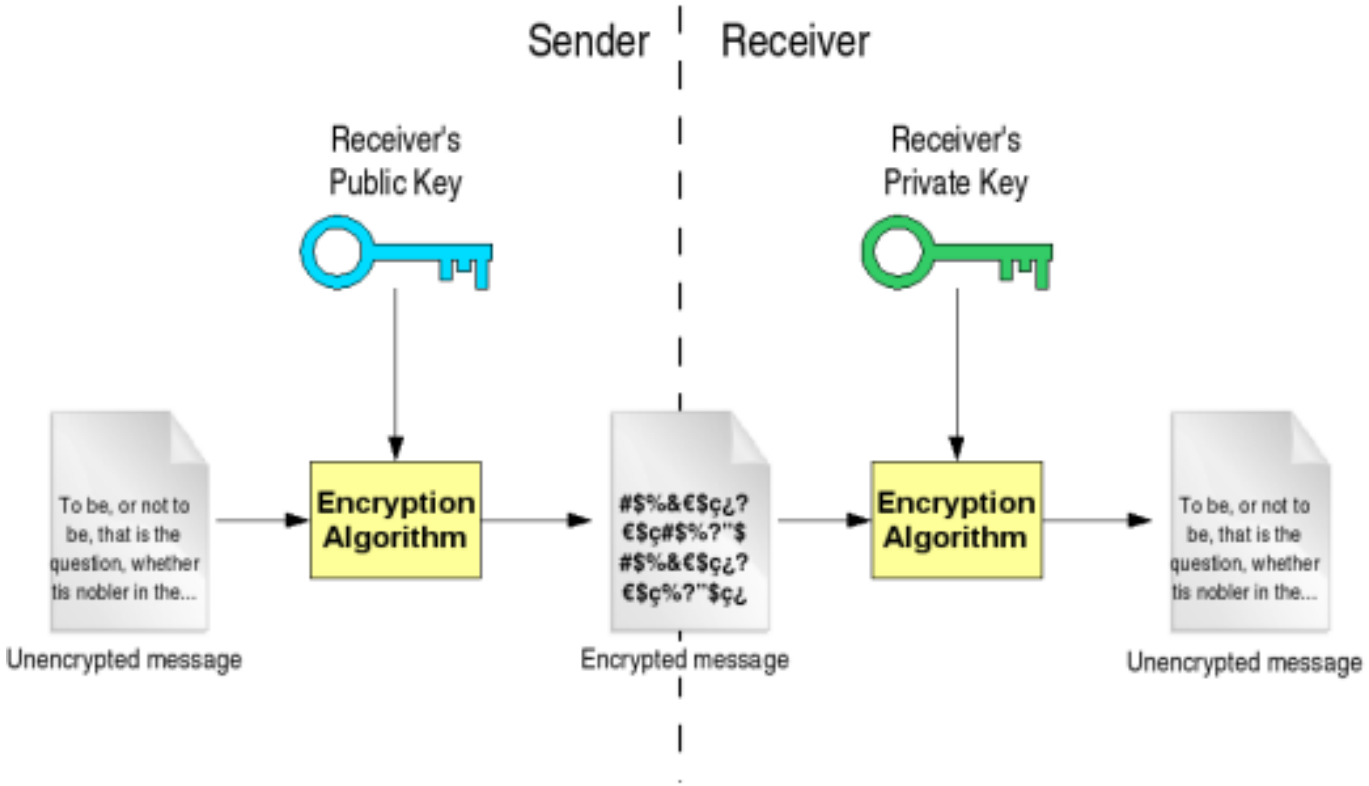
- Privacy
  - Only the sender and receiver should be able to understand the conversation
- Integrity
  - Receiving end must know that the received message was the one from the sender

Globus Security

- Certificates - Central concept in GSI
  - Information vital to identifying and authenticating user/ service
  - Distinguished Name – unique Grid id for user/service
  - "/DC=org/DC=doegrids/OU=People/CN=Raj Kettimuthu 227852”
- Certificate Authority (CA)
  - Trusted 3rd party that confirms identity
- Host credential
  - Long term credential
- User credential
  - Passphrase protected

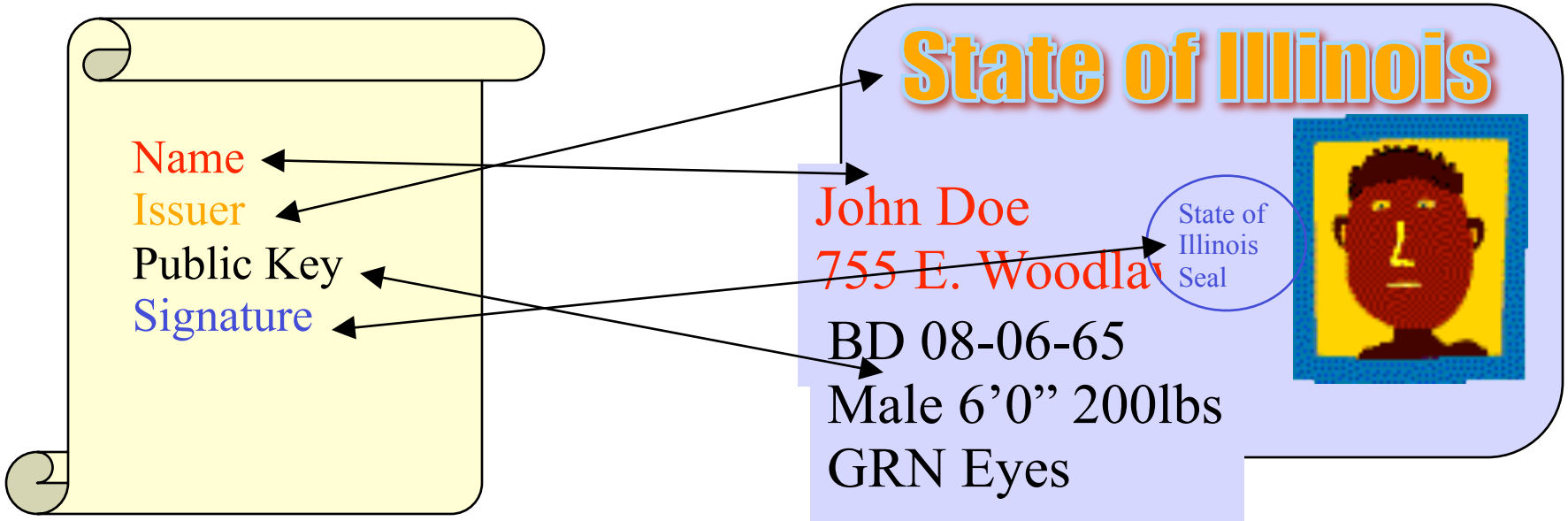
Authentication

- Private Key - known only by owner
- Public Key- known to everyone
- What one key encrypts, the other decrypts



Authentication using Digital Certificates

- Digital document that certifies a public key is owned by a particular user
- Signed by 3rd party – the Certificate Authority (CA)
- X509 standard



To know if you should trust the certificate, you have to trust the CA

Digital Signatures

Used to determine if the data has been tampered

Also, identify who signed the data

Digital signatures are generated by

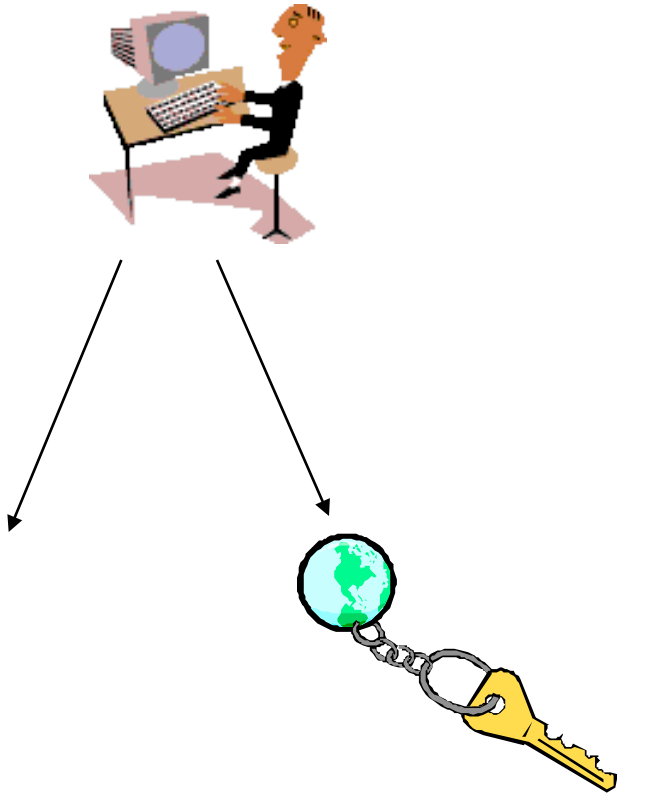
- Creating secure hash of the data
- encrypting the hash with private key

The resulting encrypted data is the signature

This hash can then be decrypted only by the corresponding public key

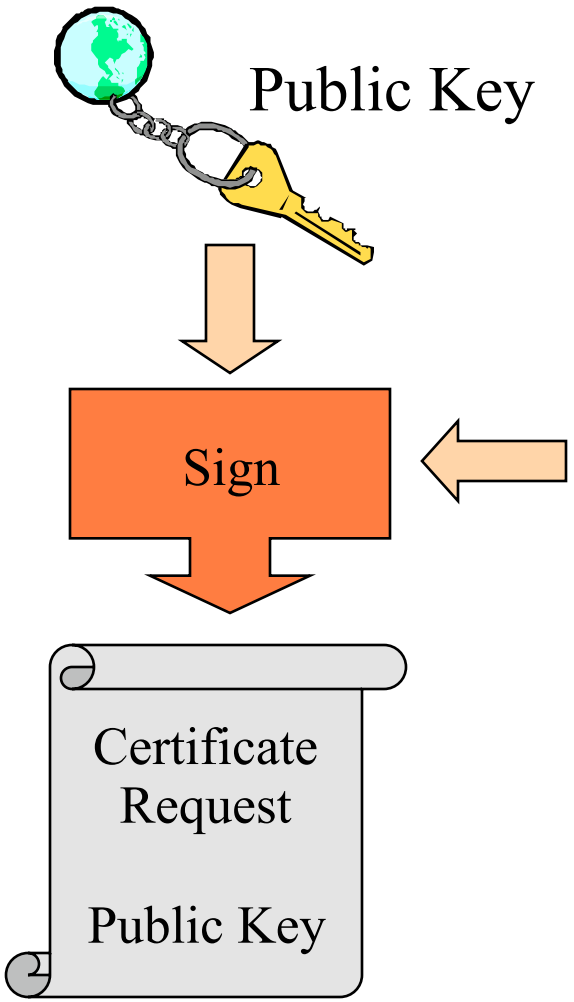
Requesting a Certificate

- To request a certificate a user starts by generating a key pair



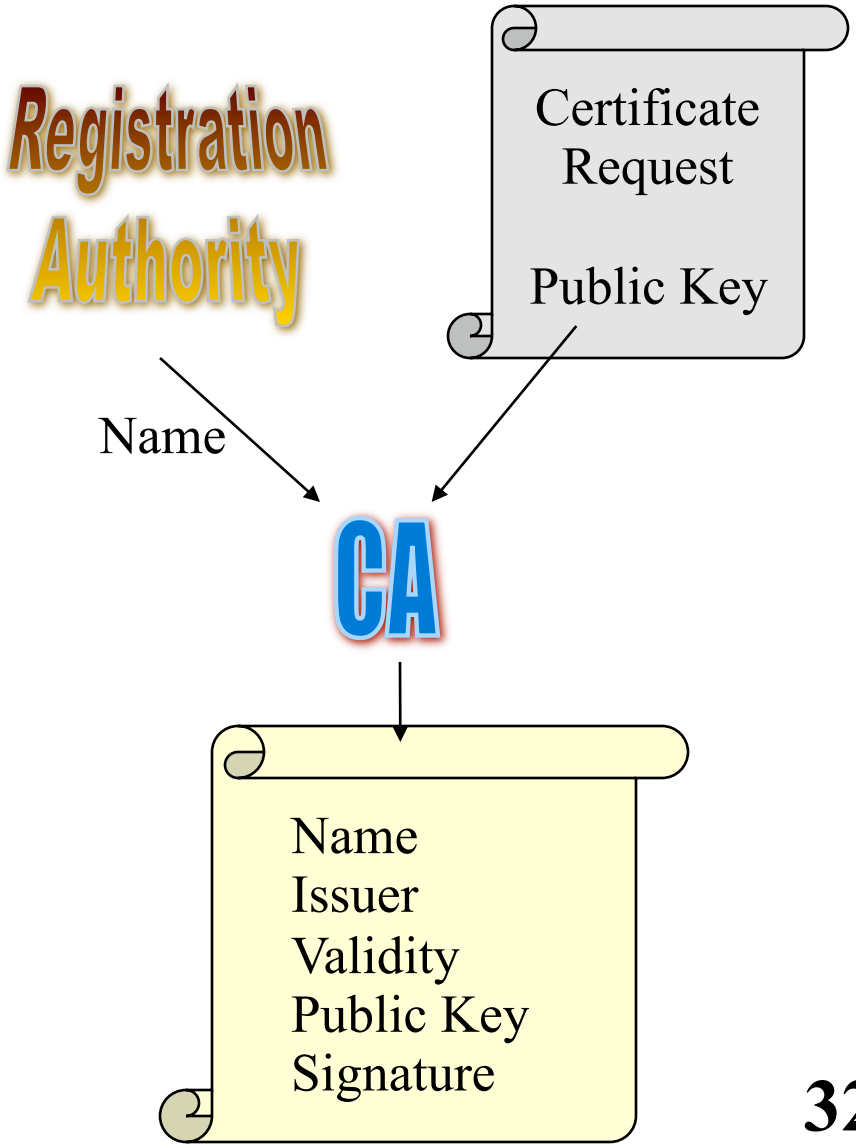
Certificate Request

- The user signs their own public key to form what is called a Certificate Request
- Email/Web upload
- Note private key is never sent anywhere



Certificate Issuance

- The CA then creates, signs and issues a certificate for the user, combining the public key and the identity



Examples of CAs

- GILDA CA
  - An online service that issues low-quality certificates. You have used GILDA CA.
- SimpleCA
  - part of Globus Toolkit. Simple to use.
- DOEgrids CA
  - production CA often used by users of the Open Science Grid
- MyProxy CA
  - part of Globus Toolkit

Proxy Certificates

- X.509 Proxy Certificates are our extension
- Standardized in IETF
- Allow for dynamic delegation
- Proxy credentials are short-lived credentials created by user
  - Proxy signed by user certificate private key
- Stored unencrypted for easy repeated access

Delegation

- Enabling another entity to run on behalf of you
- E.g Service that runs a job needs to transfer files.
- Ensure
  - Limited lifetime
  - Limited capability
- GSI uses proxy certificates for delegation

Authorization

Establishing rights of an identity

- Can user do some action on some resource

Identity based authorization

- Establish identity using authentication
- Check policy to see what identity can do
- Eg: Gridmap authorization a list of mappings from allowed DNs to user name

"/DC=org/DC=doegrids/OU=People/CN=Raj Kettimuthu 227852" kettimut

- Identity based authorization may not scale

Attribute based authorization

- Attributes are information about an entity
  - Employee of Argonne National Lab
  - Member of virtual organization ABC

GSI Stack

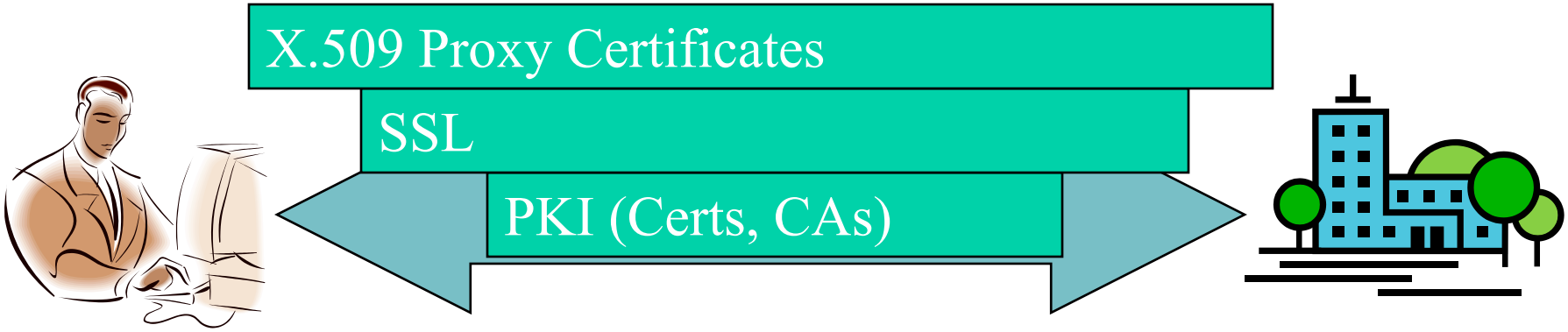
- GSI uses a standard PKI for identity certificates.
- Each entity (user, service) has an X.509 certificate from a CA that uniquely names it.

GSI Stack

- SSL, using the certificates, is used as the network protocol
  - Performs authentication, like in the web, but client as well as server
  - Also provides message protection as needed (integrity, encryption)

GSI Stack

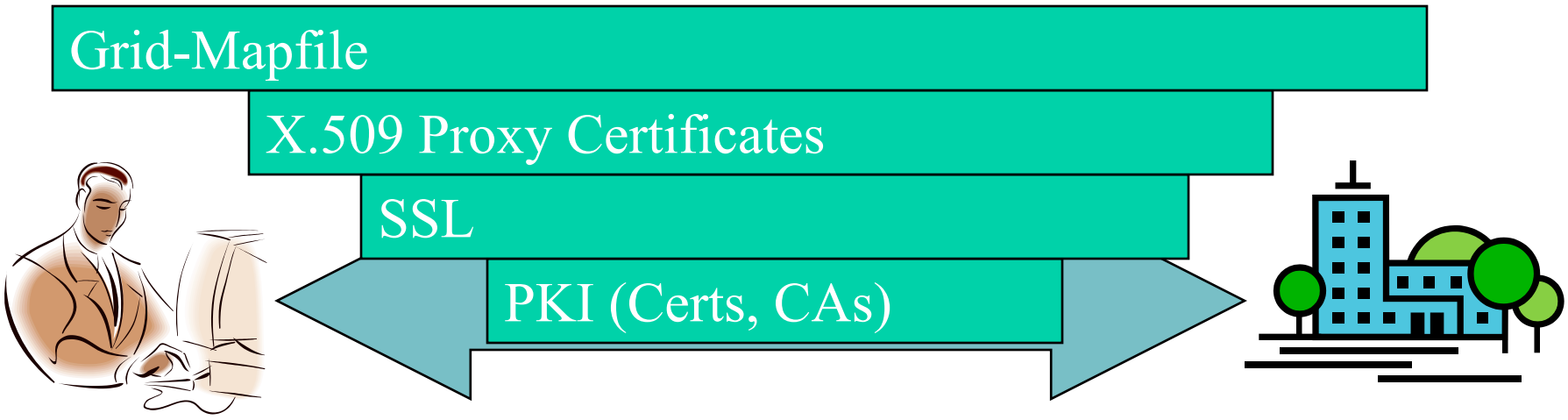
- X.509 Proxy Certificates are our extension
- Standardized in IETF (pkix)
- Allow for dynamic delegation



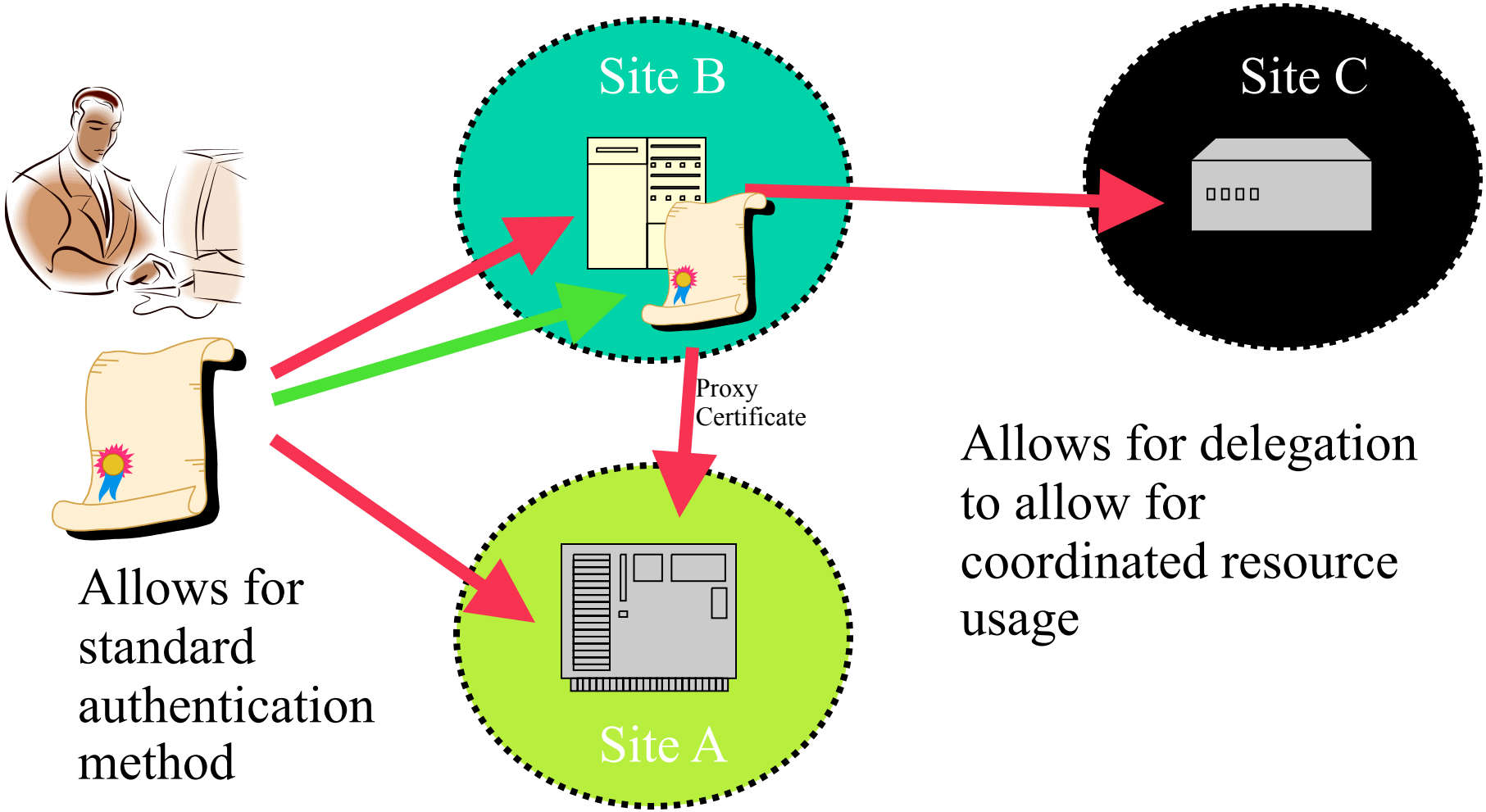
GSI Stack

• Grid-Mapfile maps Grid users (identified by certificates) to local users (e.g. Unix account)

• Allows authorization using normal local methods (e.g. filesystem perms, quotas)



GSI-Enabled Coordination

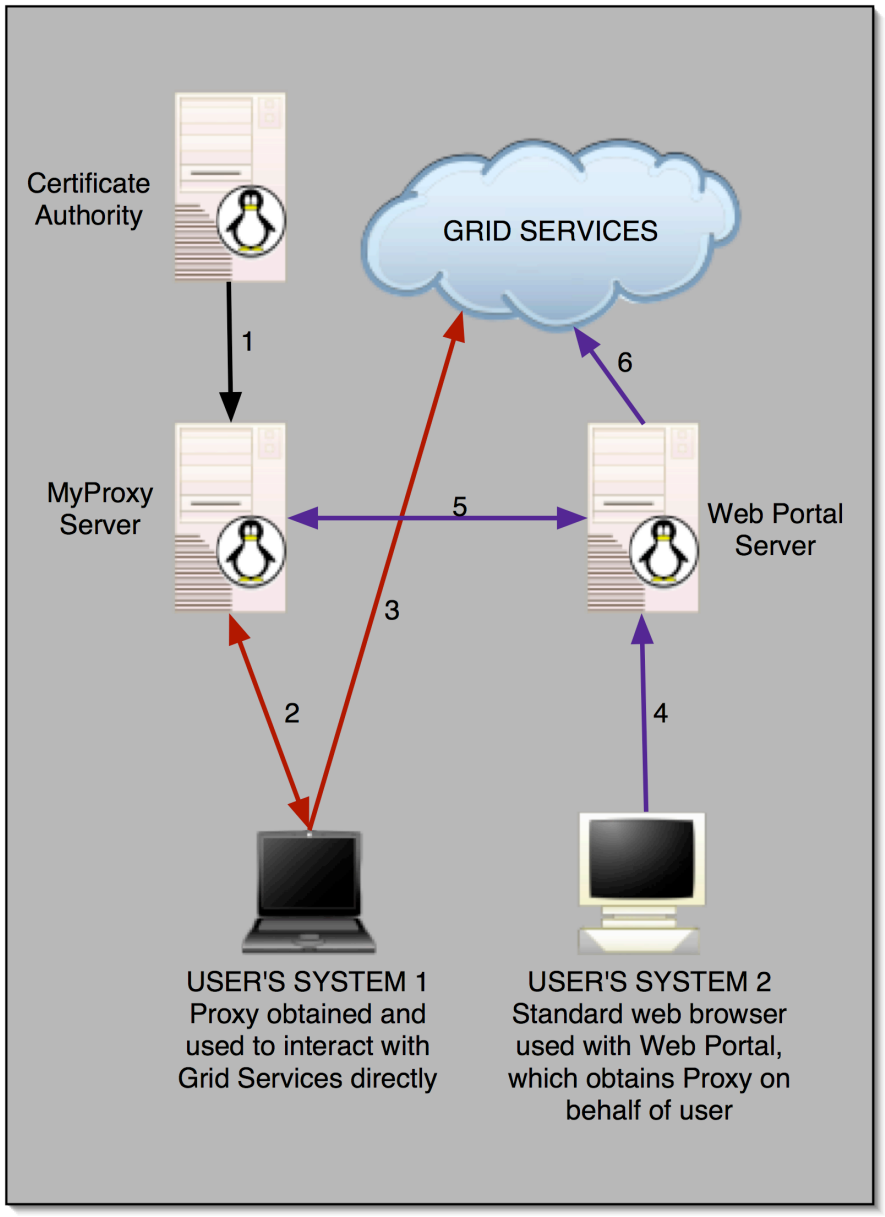


Allows for standard authentication method

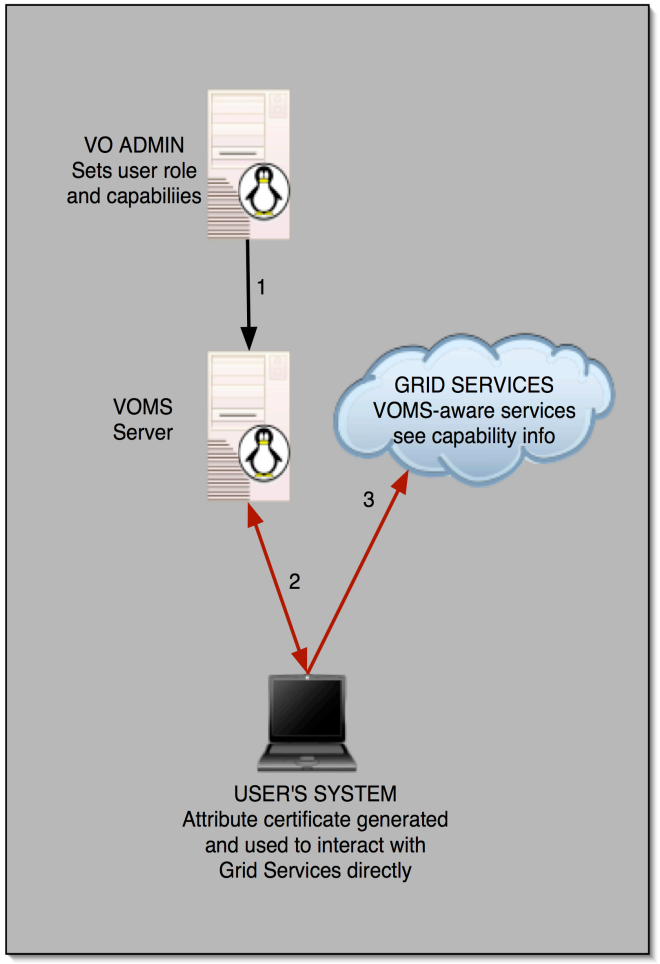
Allows for delegation to allow for coordinated resource usage

Globus and Delegation: MyProxy

- Remote service that stores user credentials
  - Users request proxies for local use
  - Web Portals request user proxies for use with back-end Grid services
- Grid administrators can pre-load credentials in the server for users to retrieve when needed

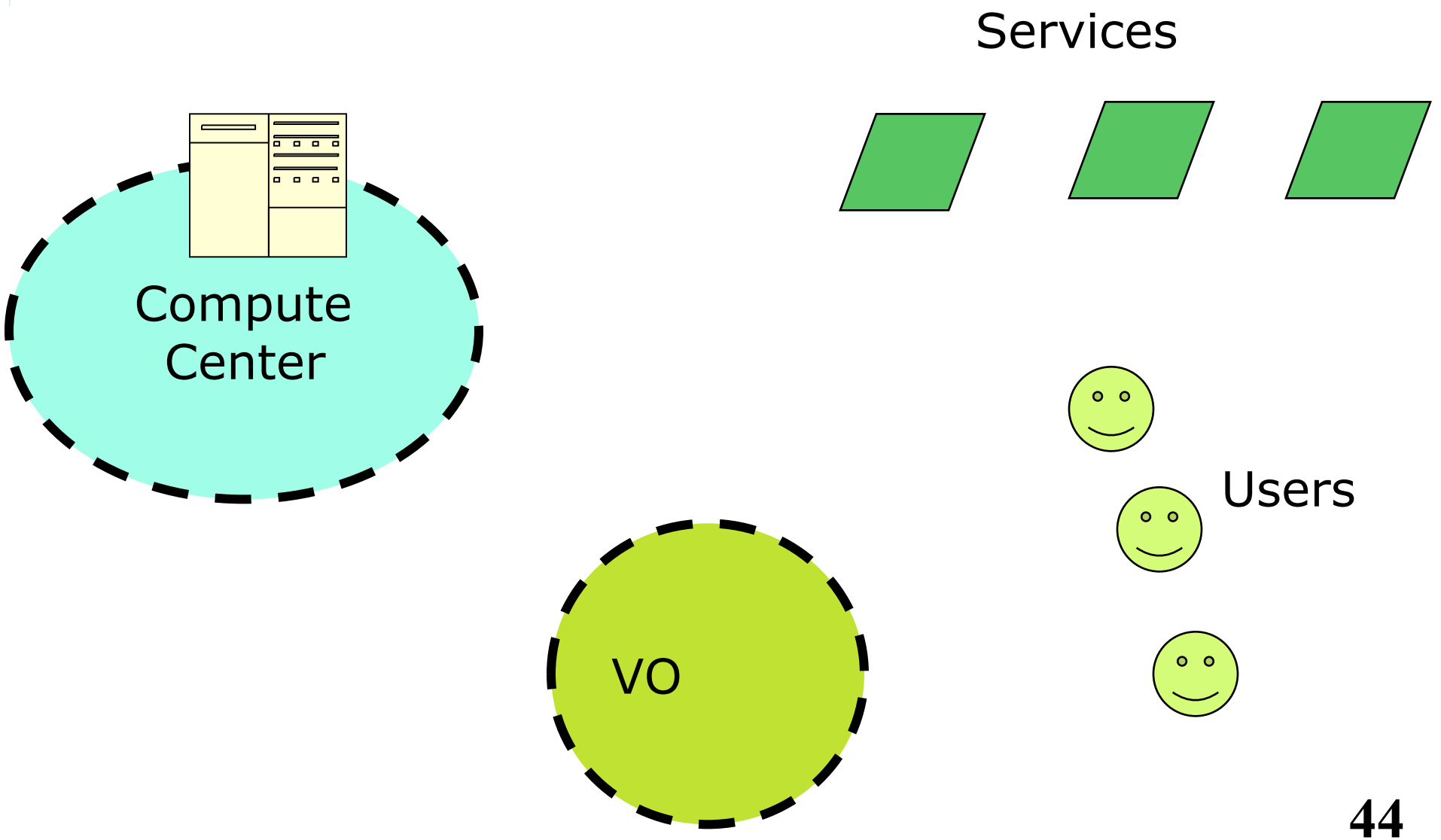


VOMS

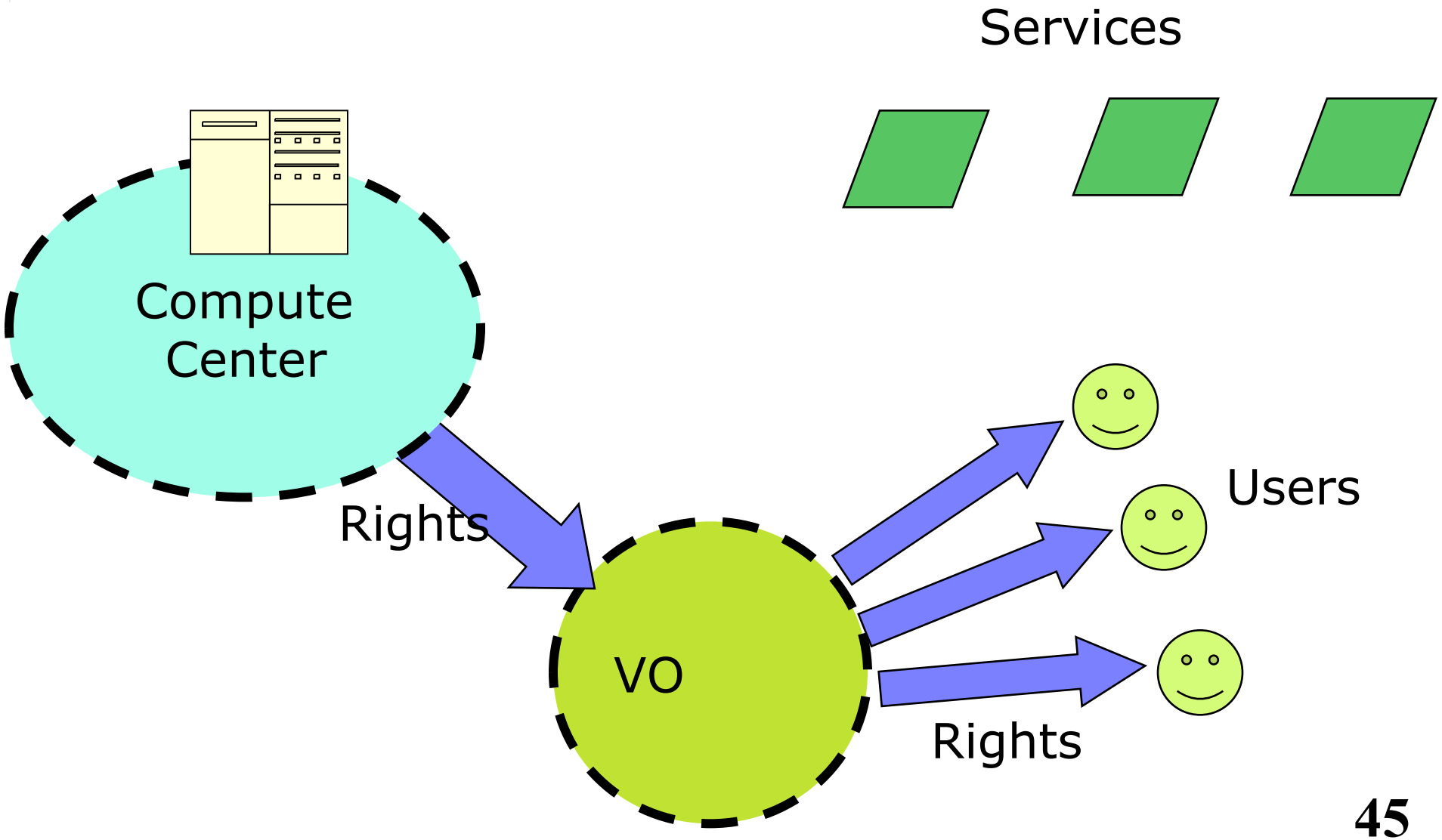


- A community-level group membership system
- Database of user roles
  - Administrative tools
  - Client interface
- voms-proxy-init
  - Uses client interface to produce an attribute certificate (instead of proxy) that includes roles & capabilities signed by VOMS server
  - Works with non-VOMS services, but gives more info to VOMS-aware services
- Allows VOs to centrally manage user roles

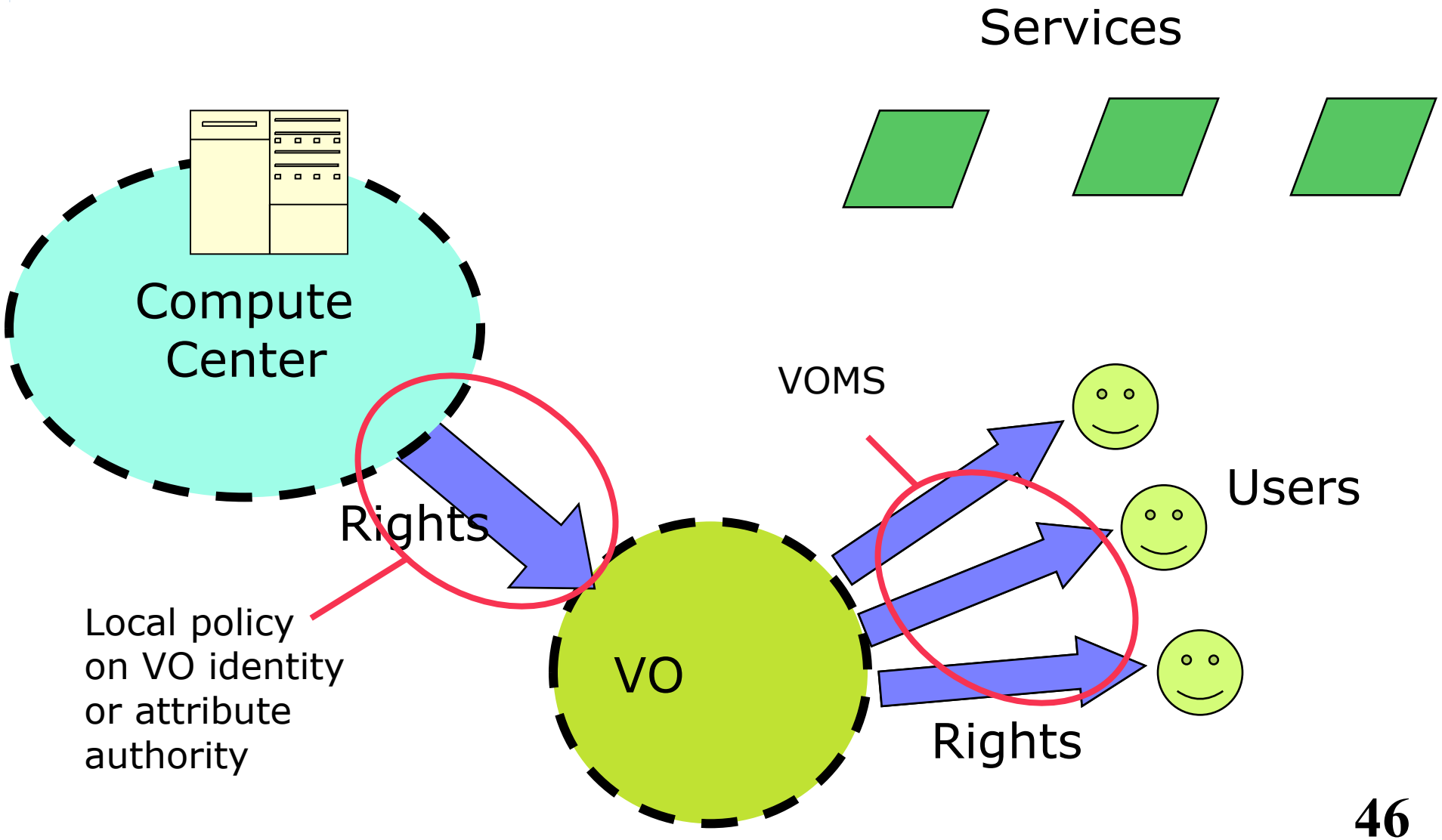
Globus Security: How It Works



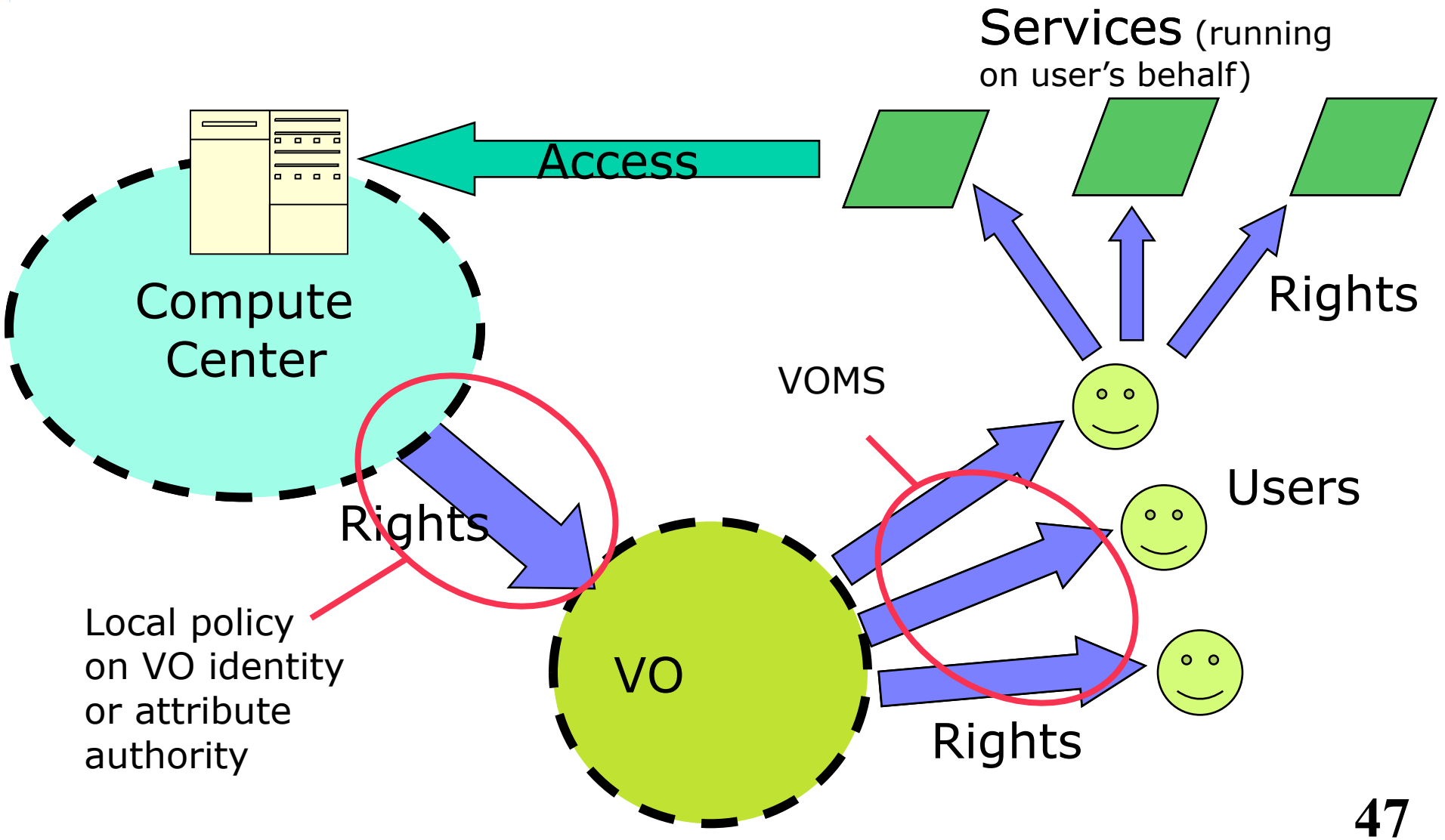
Globus Security: How It Works



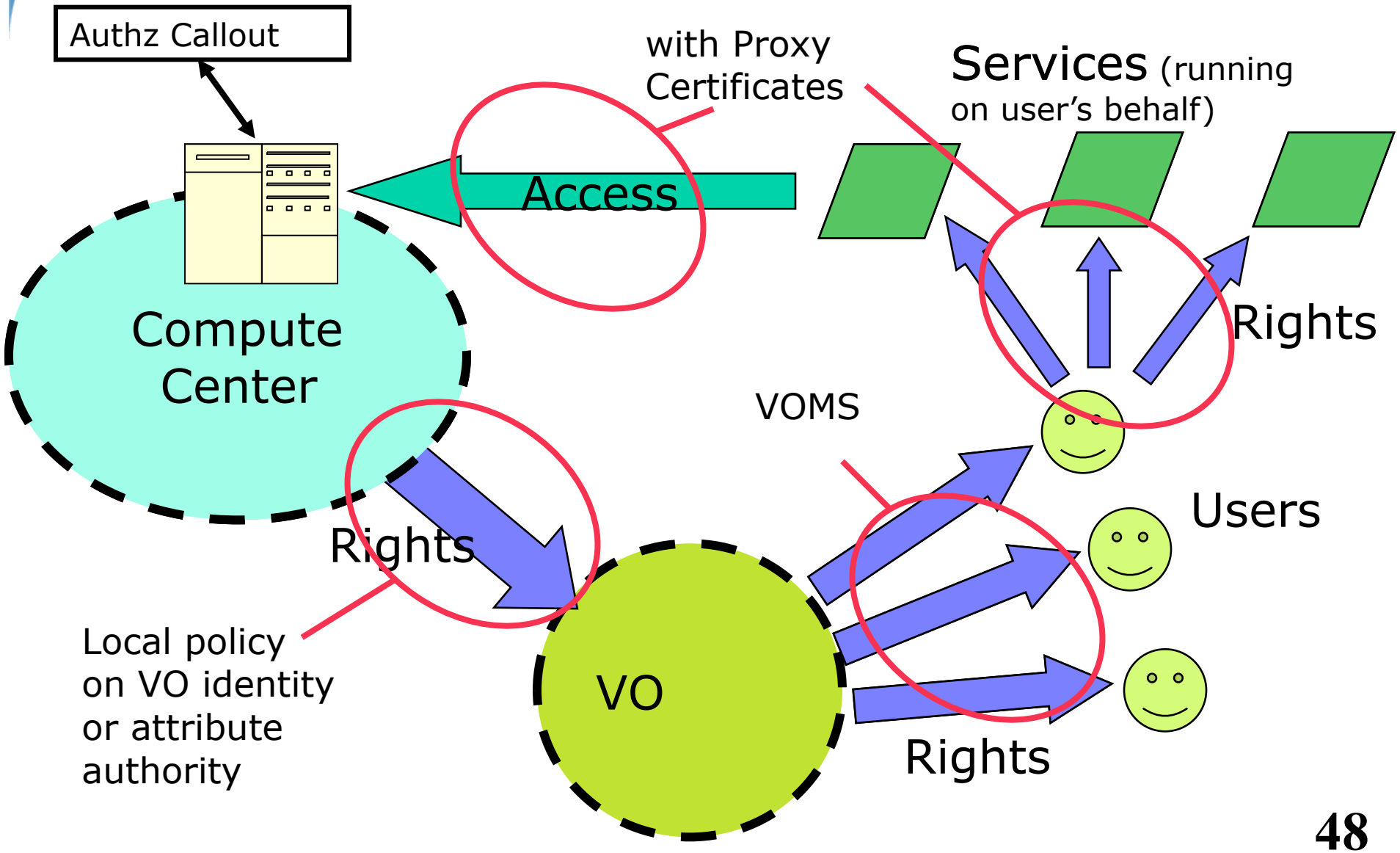
Globus Security: How It Works



Globus Security: How It Works



Globus Security: How It Works



A Cautionary Note

- Grid security mechanisms are tedious to set up
  - If exposed to users, hand-holding is usually required
  - These mechanisms can be *hidden* from end users, but still used behind the scenes
- These mechanisms exist for good reasons.
  - Many useful things can't be done without Grid security
  - It is unlikely that an ambitious project could go into production operation without security like this
  - Most successful projects end up using Grid security, but using it in ways that end users don't see much

More Specifically, I May Want To …

- Manage who is allowed to access my service (or my experimental data or …)
- Ensure reliable & secure distribution of data from my lab to my partners
- Run 10,000 jobs on whatever computers I can get hold of

File Management

- **Stage/move** large data to/from nodes
  - GridFTP for basic file movement
- **Locate** data of interest
  - Replica Location Service (RLS)

What is GridFTP?

- High-performance, reliable data transfer protocol optimized for high-bandwidth wide-area networks
- Based on FTP protocol - defines extensions for high-performance operation and security
- Standardized through Open Grid Forum (OGF)
- GridFTP is the OGF recommended data movement protocol

the globus alliance
www.globus.org

GridFTP

- We (Globus Alliance) provide a reference implementation:
  - Server
  - Client tools (globus-url-copy)
  - Development Libraries
- Multiple independent implementations can interoperate
  - Fermi Lab and U. Virginia have home grown servers that work with ours

Globus GridFTP

Performance

- Parallel TCP streams, optimal TCP buffer
- Non TCP protocol such as UDT

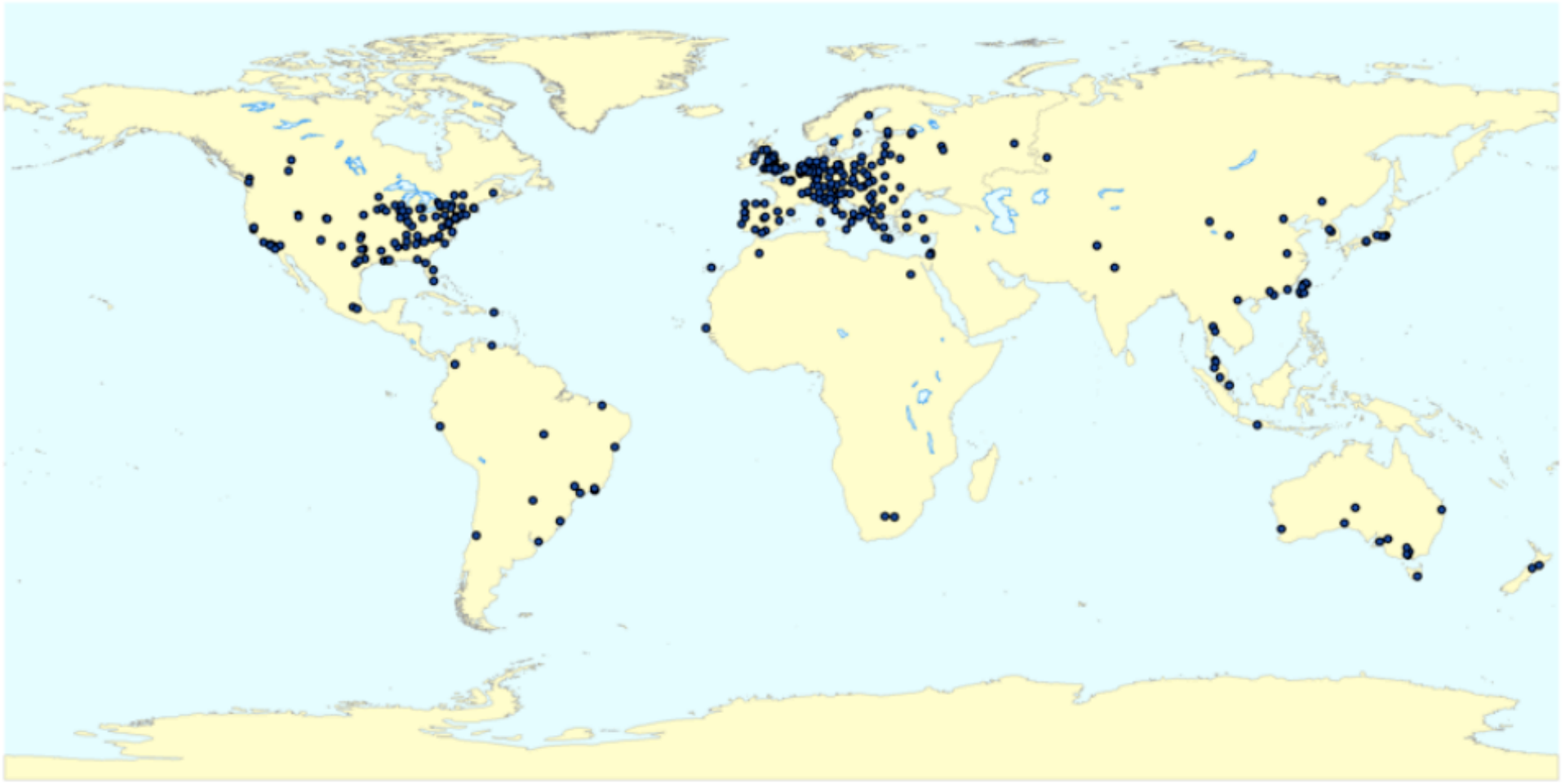
Cluster-to-cluster data movement

Multiple security options

- Anonymous, password, SSH, GSI

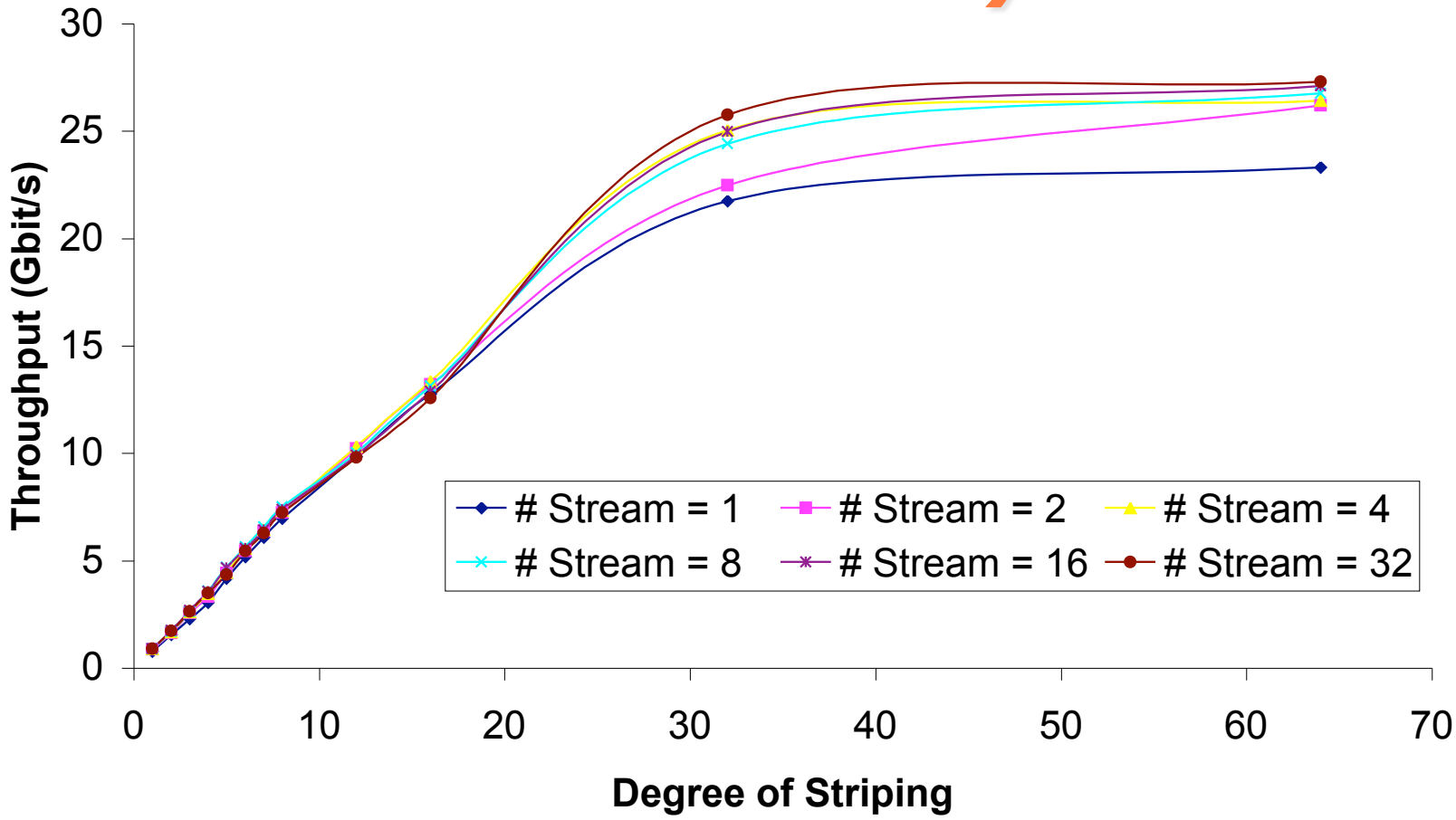
Support for reliable and restartable transfers

GridFTP Servers Around the World



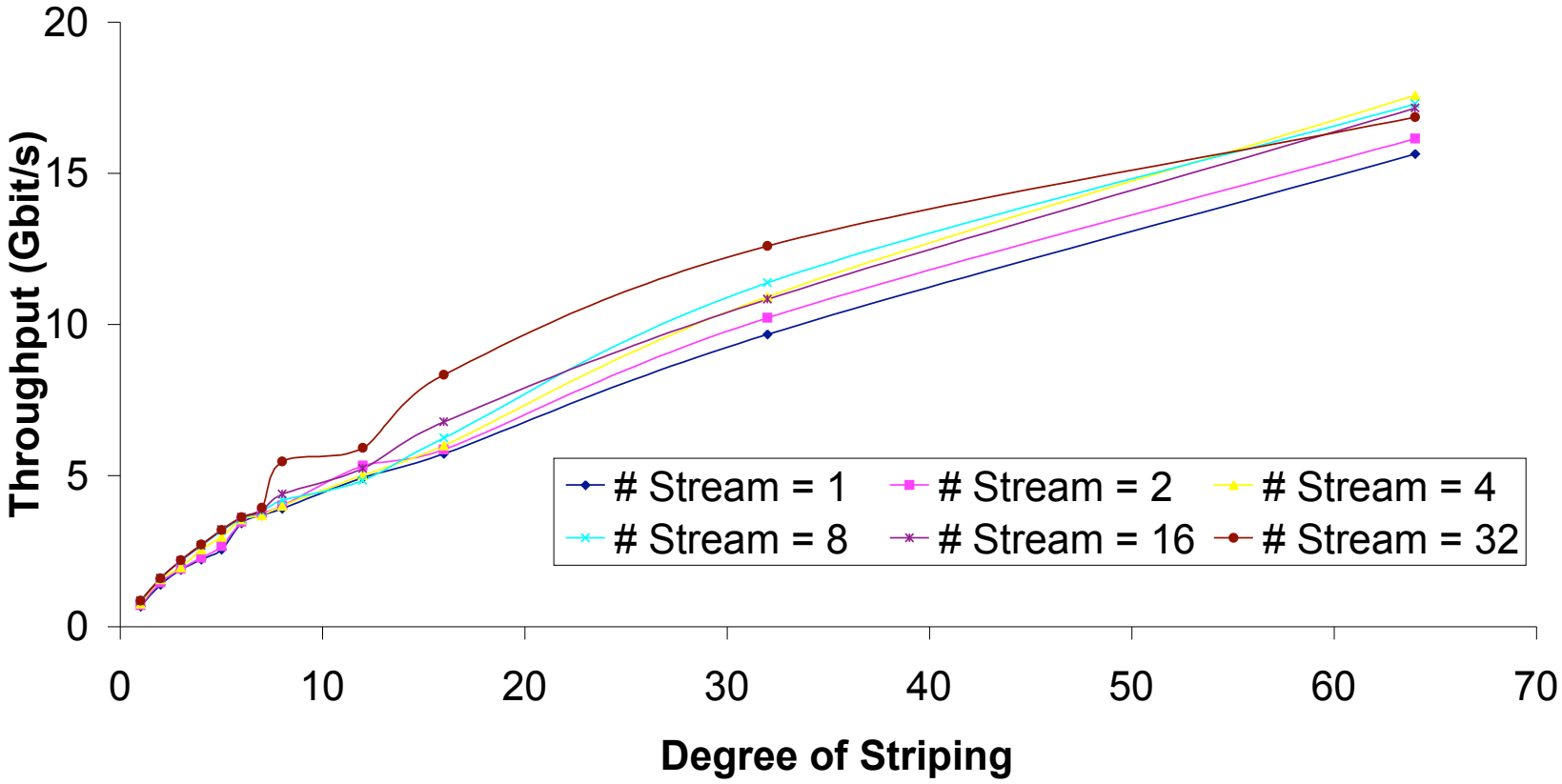
Created by Tim Pinkawa (Northern Illinois University) using MaxMind's GeoIP technology (http://www.maxmind.com/app/ip-locate).

Memory to Memory over 30 Gigabit/s Network (San Diego — Urbana)



the globus alliance
www.globus.org

Disk to Disk over 30 Gigabit/s Network (San Diego — Urbana)



Understanding GridFTP

- Two channel protocol like FTP
- Control Channel
  - Command/Response
  - Used to establish data channels
  - Basic file system operations eg. mkdir, delete etc
- Data channel
  - Pathway over which *file* is transferred
  - Many different underlying protocols can be used
    - MODE command determines the protocol

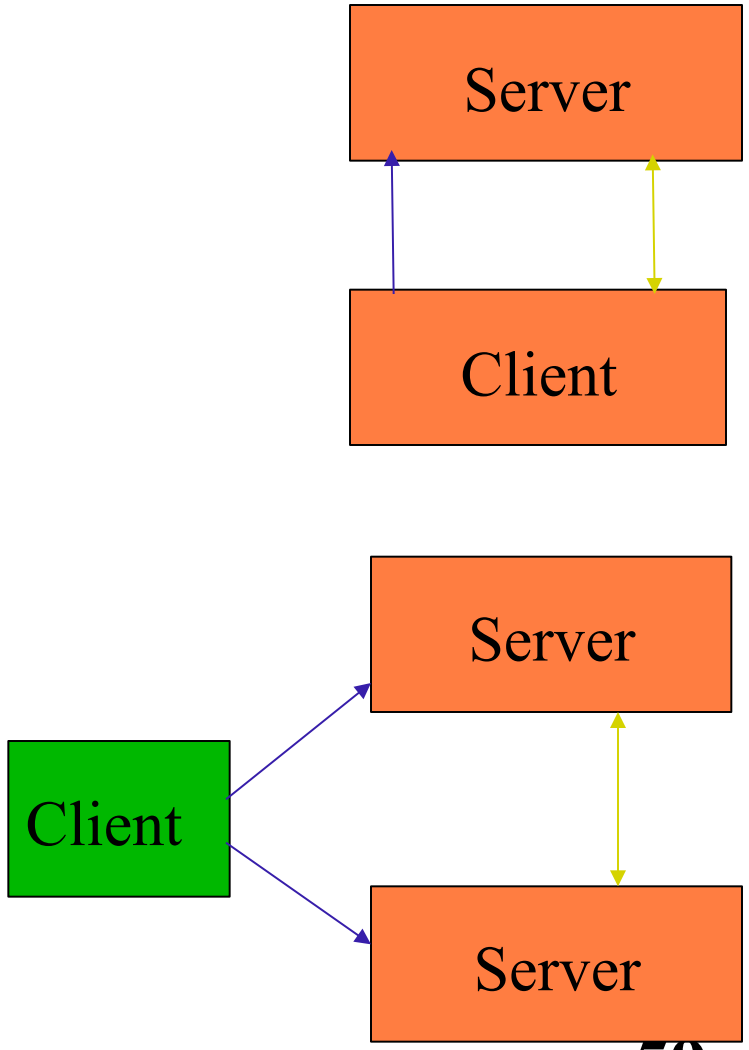
Client/Server and 3rd Party Transfers

Two party transfer

- The client connects and forms a CC with the server
- Information is exchanged to establish the DC
- A file is transferred over the DC

Third party transfer

- Client initiates data transfer between 2 servers
- Client forms CC with 2 servers.
- Information is routed through the client to establish DC between the two servers.
- Data flows directly between servers
- Client is notified by each server SPI when the transfer is complete



Control Channel Establishment

Server listens on a well-known port (2811)

Client form a TCP Connection to server

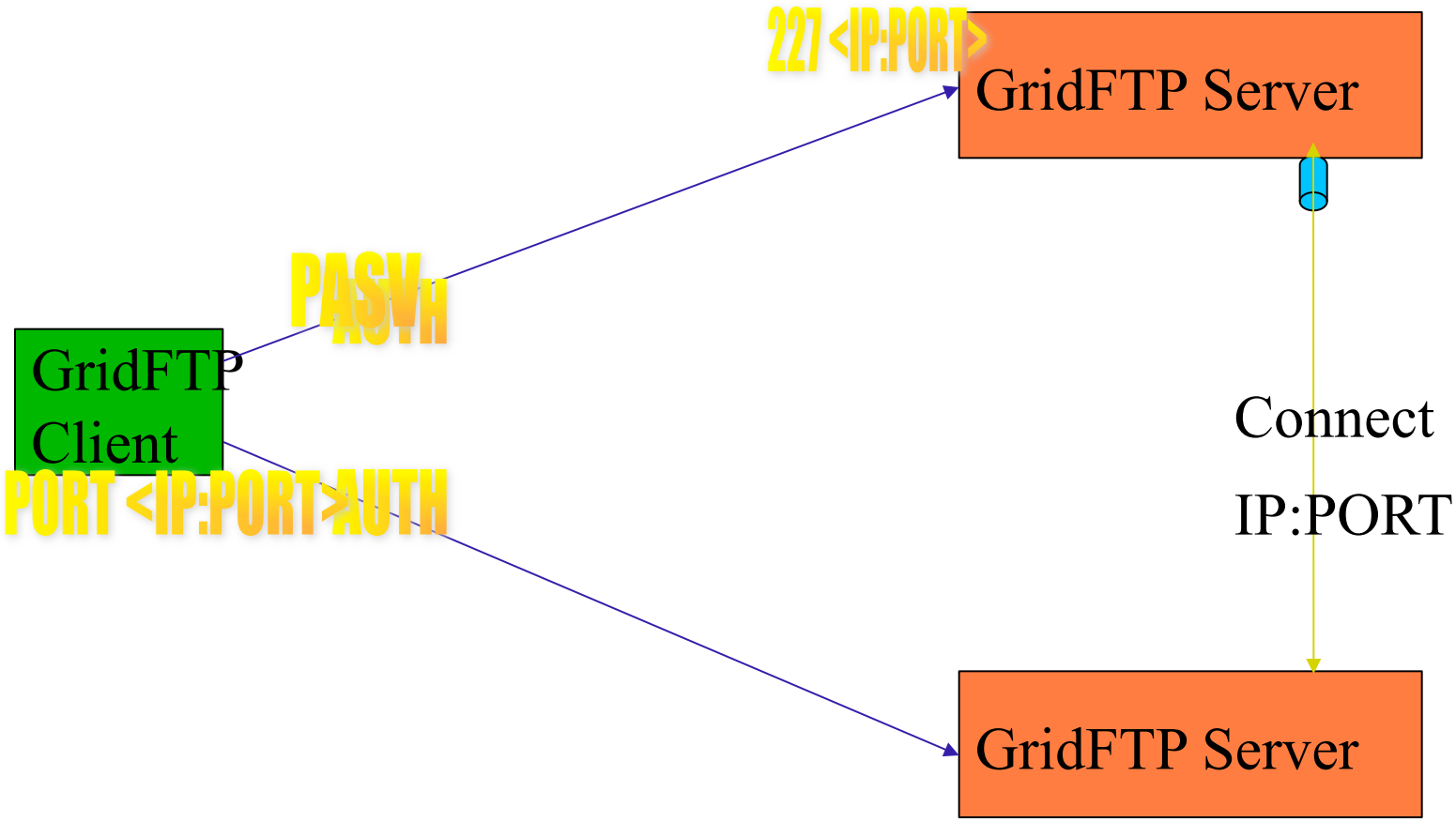
220 banner message

Authentication

- Anonymous
- Clear text USER <username>/PASS <pw>
- Base 64 encoded GSI handshake

230 Accepted/530 Rejected

Data Channel Establishment



Data Channel Protocols

MODE Command

Allows the client to select the data channel protocol

MODE S

Stream mode, no framing

Legacy RFC959

MODE E

GridFTP extension

Parallel TCP streams

Data channel caching

Descriptor (8 bits)	Size (64 bits)	Offset (64 bits)

Globus-url-copy

- Command line scriptable client
- Globus does not provide an interactive client
- Commonly used client for GridFTP
- Syntax overview
  - globus-url-copy [options] srcURL dstURL
  - guc gsiftp://localhost/foo file:///bar
    - Client/server, using FTP stream mode
  - guc –vb –dbg –tcp-bs 1048576 –p 8 gsiftp://localhost/foo gsiftp://localhost/bar
    - 3rd party transfer, MODE E
- URL rules
  - protocol://[user:pass@][host]/path
  - host can be anything resolvable - IP address, localhost, DNS name

Security Options

Clear text (RFC 959)

- Username/password
- Anonymous mode (anonymous/<email addr>)
- Password file

SSHFTP

- Use ssh/sshd to form the control connection

GSIFTP

- Authenticate control and data channels with GSI

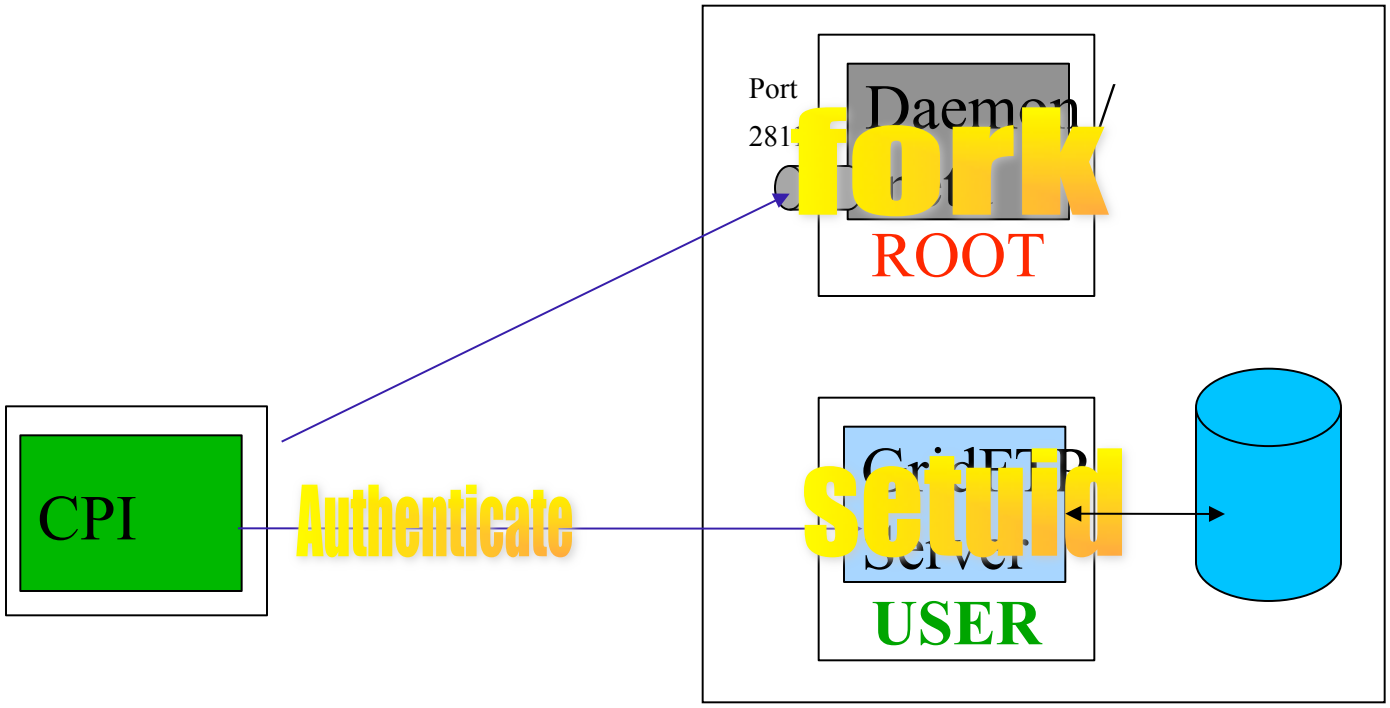
User Permissions

User is mapped to a local account and file permissions are handled by the OS

inetd or daemon mode

- Daemon mode - GridFTP server is started by hand and listens for connections on port 2811
- Inetd/xinetd - super server daemon that manages internet services
- Inetd can be configured to start up a GridFTP server upon receiving a connection on port 2811

inetd/daemon Interactions



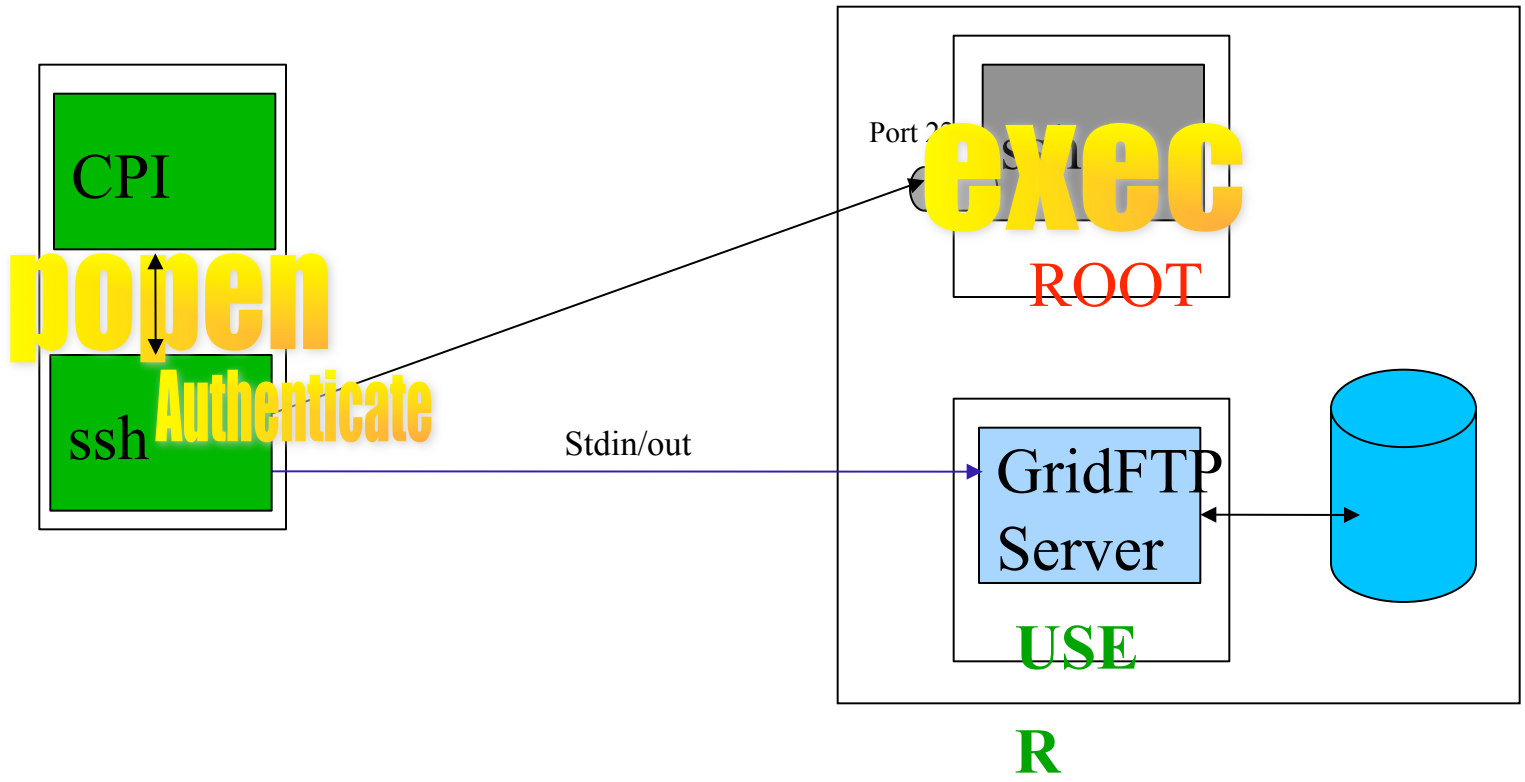
GridFTP Over SSH

sshd acts similar to inetd

control channel is routed over ssh

- globus-url-copy *popens* ssh
- ssh authenicates with sshd
- ssh/sshd remotely starts the GridFTP server as user
- stdin/out becomes the control channel

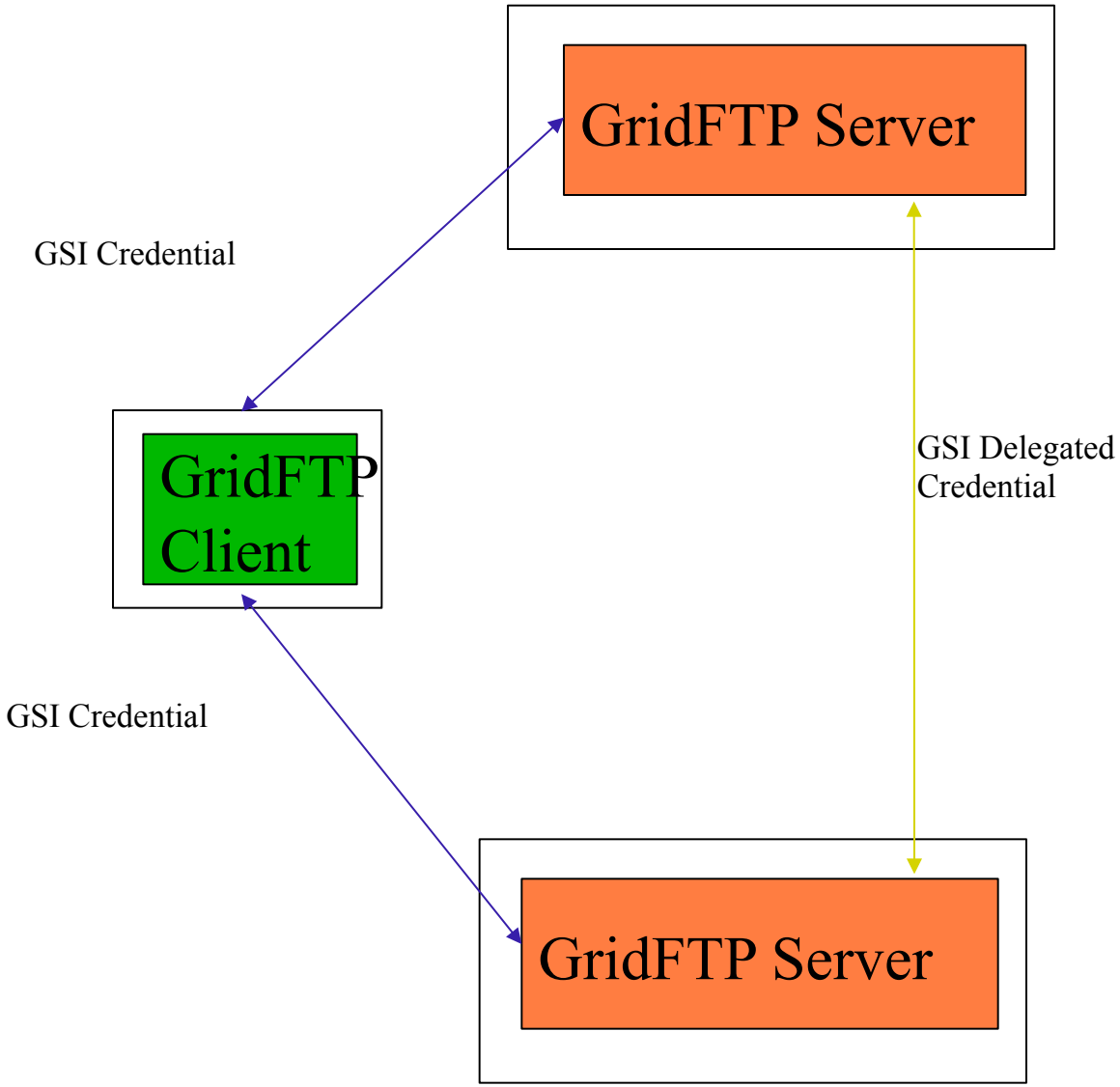
sshftp:// Interactions



GSI Authentication

- Strong security on both channels
  - SSH does not give us data channel security
- Delegation
  - Authenticates DC on clients behalf
  - Flexibility for grid services such as RFT
    - Agents can authenticate to GridFTP servers on users behalf
  - Enables encryption, integrity on data channel

GSI Authentication



Troubleshooting

Can I get connected?

telnet to the port: telnet hostname port

2811 is the default port

You should get something like this:

<add GridFTP banner>

If not, you have firewall problems, or server config problems.

Setting TCP buffer sizes

It is critical to use the optimal TCP send and receive socket buffer sizes for the link you are using.

- Recommended size to fill the pipe
  - 2 x Bandwidth Delay Product (BDP)
- Recommended size to leave some bandwidth for others
  - around 20% of (2 x BDP) = .4 * BDP

Setting TCP buffer sizes

- Default TCP buffer sizes are way too small for today's high speed networks
  - Until recently, default TCP send/receive buffers were typically 64 KB
  - tuned buffer to fill Argonne to LBL link**: 8 MB**
    - 125X bigger than the default buffer size
  - with default TCP buffers, you can only get a small % of the available bandwidth!

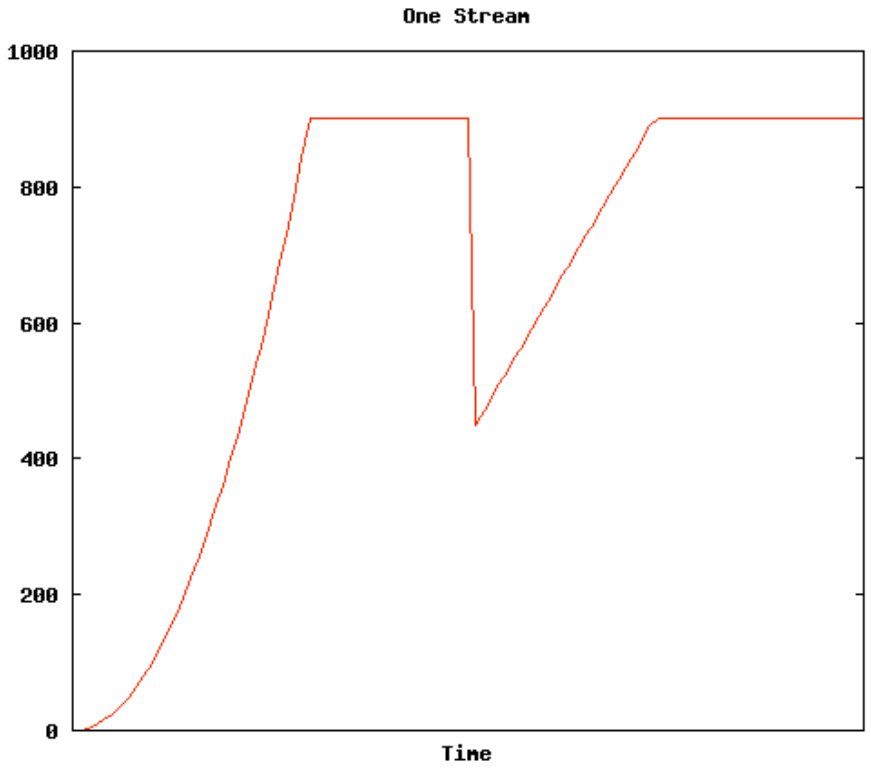
TCP tuning

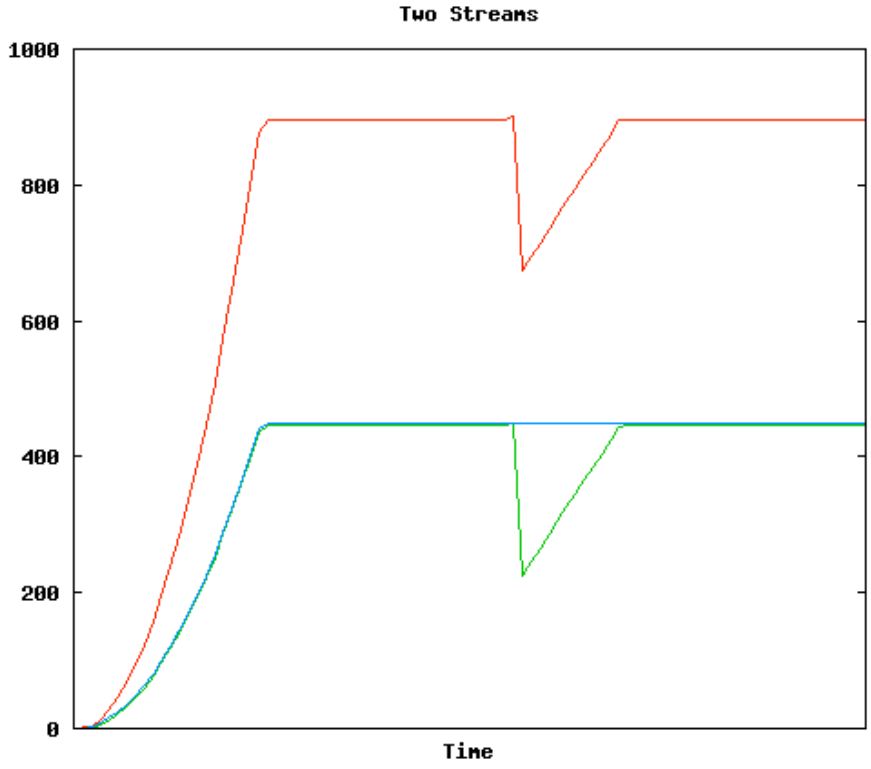
- Many OS's now include TCP autotuning
  - TCP send buffer starts at 64 KB
  - As the data transfer takes place, the buffer size is continuously re-adjusted up to max autotuning size
- Default autotuning maximum buffers on Linux 2.6: 256K to 1MB, depending on version

```
net.core.rmem_max = 16777216
net.core.wmem_max = 16777216
# autotuning min, default, and max number of bytes to use
net.ipv4.tcp_rmem = 4096 87380 16777216
net.ipv4.tcp_wmem = 4096 65536 16777216
```

- http://fasterdata.es.net/TCP-tuning/

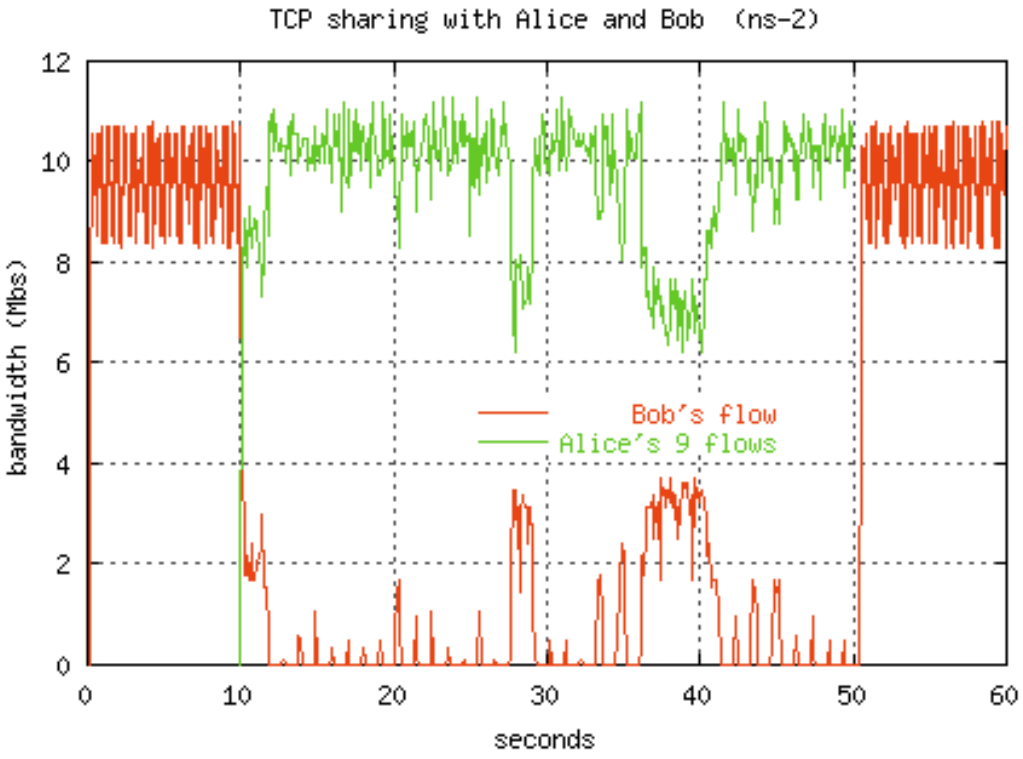
Parallel Streams





Parallel TCP Streams

- Potentially unfair
- Reduces the severity of a congestion event
  - Only effects 1/p of the overall transfer
- Faster recovery
  - Smaller size to recover
- But they are necessary when
  - you don't have root access
  - and can't convince the sysadmin to increase the max TCP buffers



graph from Tom Dunigan, ORNL

Data channel caching

Establishing a data channel can be expensive

- Round trips over high latency links
- Security handshake can be expensive

Mode E introduces data channel caching

- Mode S closes the connection to indicate end of data
- Mode E uses meta data to indicate file barriers
  - Doesn't need to close

Descriptor (8 bits)	Size (64 bits)	Offset (64 bits)

Data Channel Protocols

MODE Command

Allows the client to select the data channel protocol

MODE S

Stream mode, no framing

Legacy RFC959

MODE E

GridFTP extension

Parallel TCP streams

Data channel caching

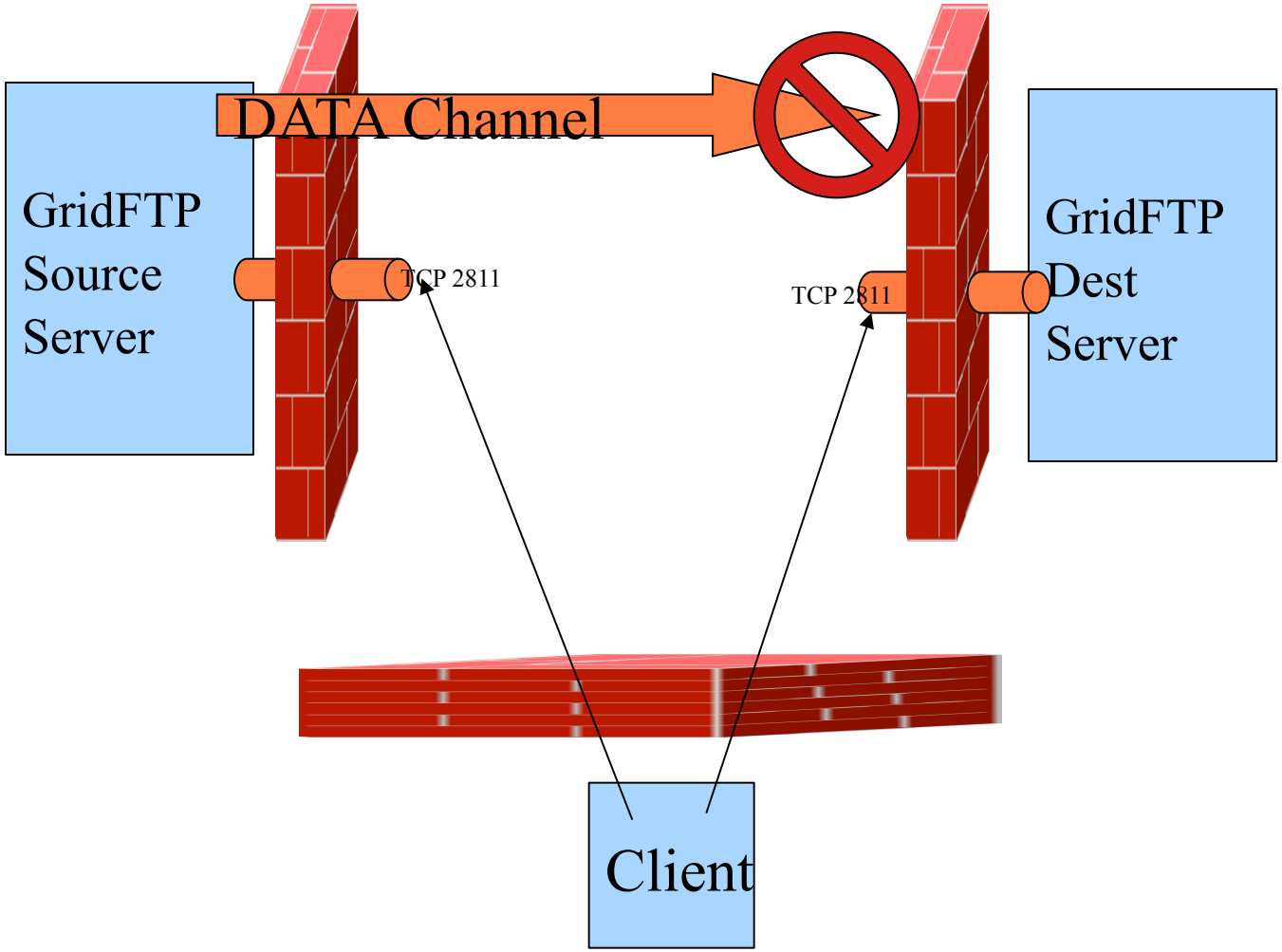
Descriptor (8 bits)	Size (64 bits)	Offset (64 bits)

Firewall

- Control channel port is statically assigned
- Data channel ports dynamically assigned
- Mode E requires that the data sender make an active connection

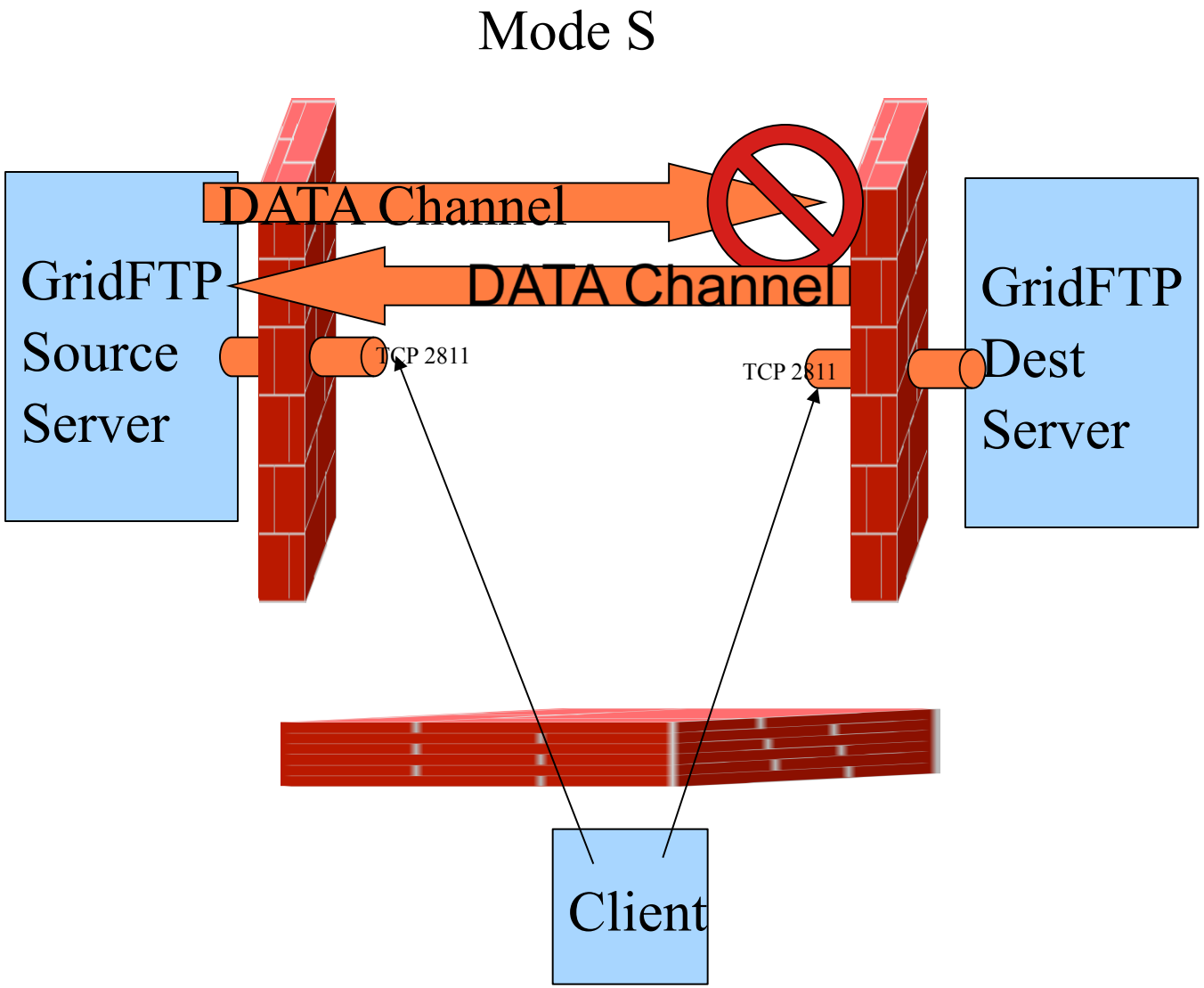
Firewall

- Outgoing allowed at sender, incoming blocked at receiver



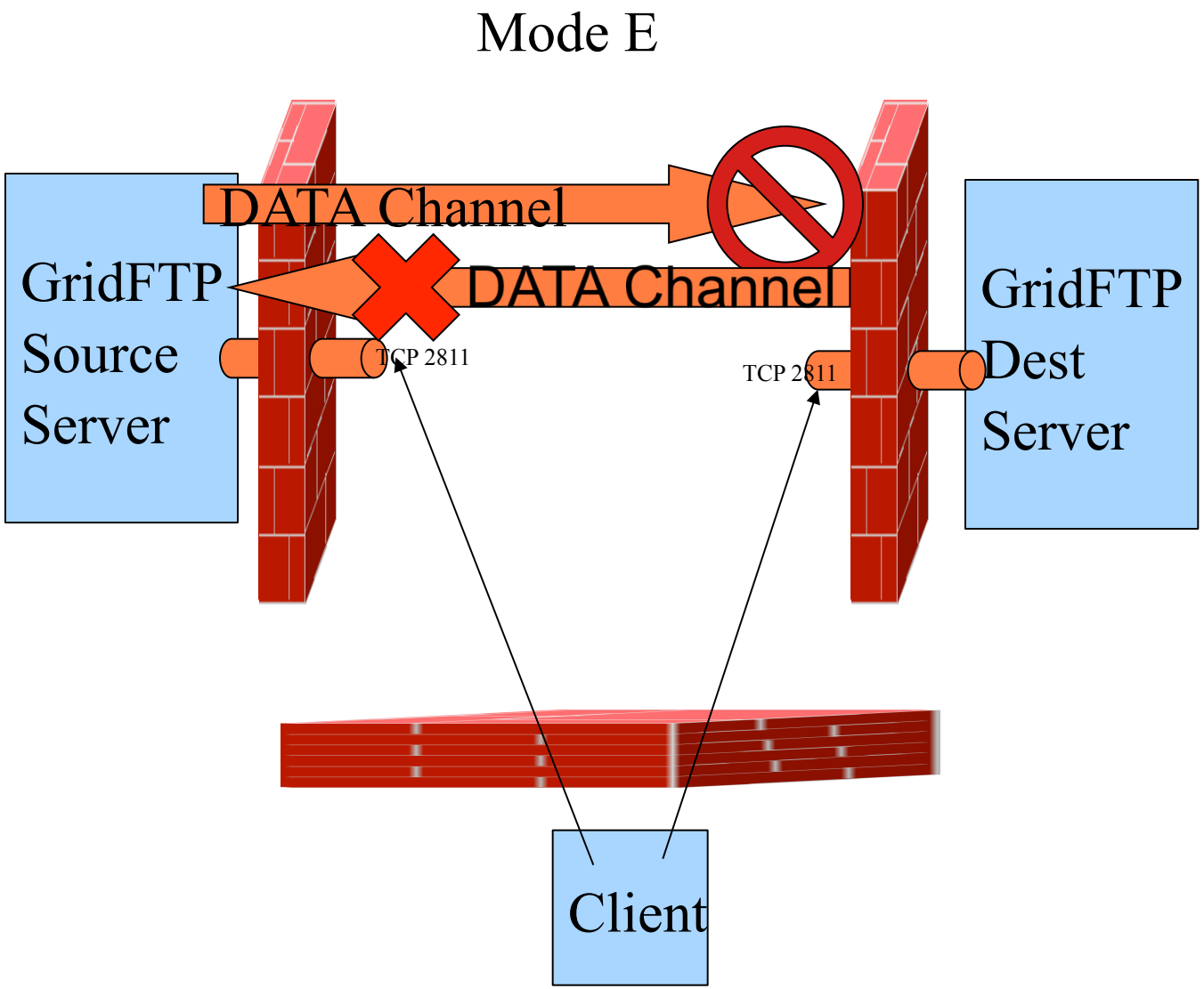
Firewall

- Outgoing allowed at sender, incoming blocked at receiver



Firewall

- Outgoing allowed at sender, incoming blocked at receiver



Firewall

- Open a port range on the receiver's ends firewall and set GLOBUS_TCP_PORT_RANGE to that open range
- 50000-51000 is the recommended port range for data channel connections
- export GLOBUS_TCP_PORT_RANGE = 50000,51000

Firewall

Outgoing blocked at sender

- Can open a range of ports for outgoing connections to specific set of remote hosts (any remote port)
- Use GLOBUS_TCP_SOURCE_RANGE to make the local end bound to a specified range
- If outgoing connections can be opened up only for specific remote port range at specific remote hosts
  - firewall rule needs to modified on a case-by-case basis

Partial File Transfer

Large file transfer fails

- We don't want to start completely over
- Ideally we start where we left off

Restart markers sent periodically

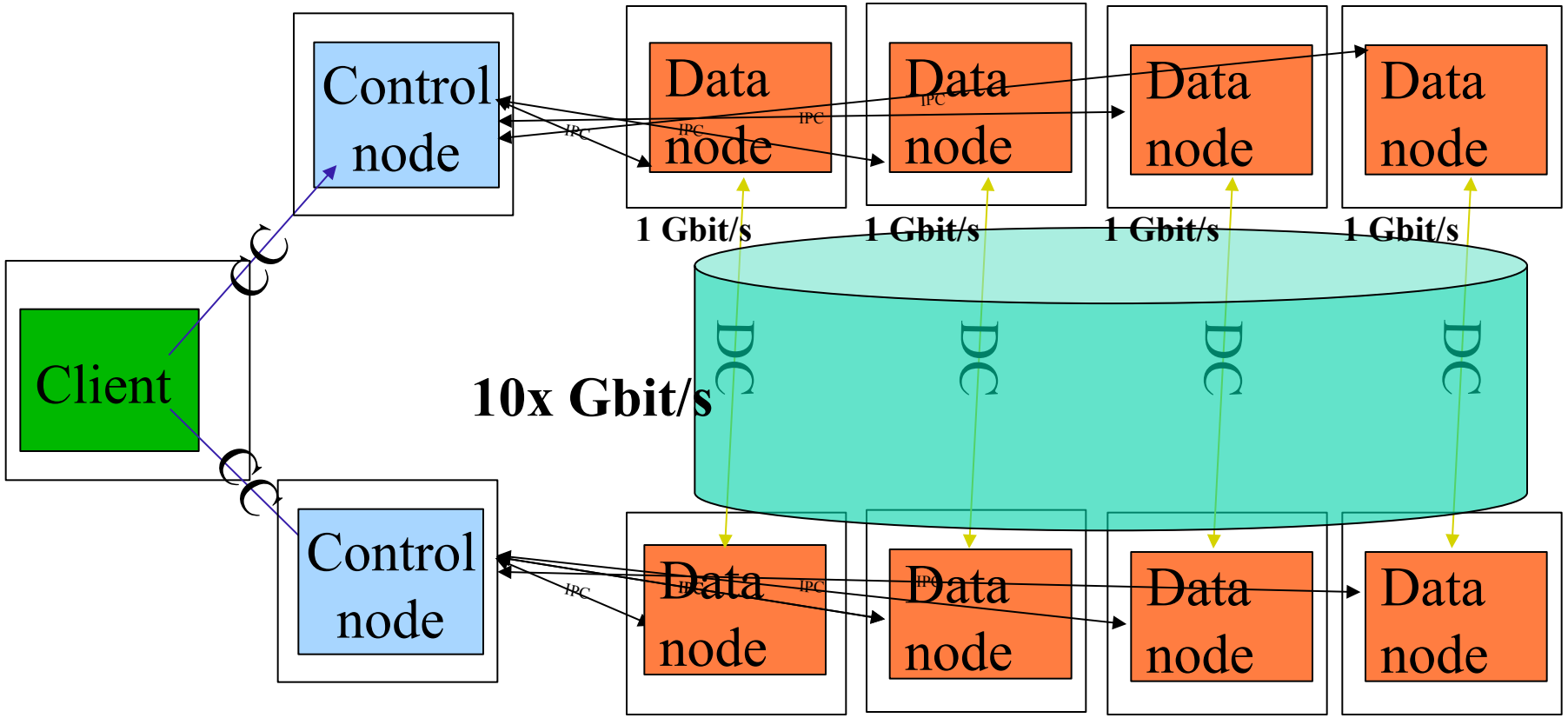
- Contain blocks written to disk
- Sent every 5s by default
- In worst case recovery sends 5s of redundant data

Striping or Cluster-to-cluster transfer

A coordinated transfer between multiple nodes at end of the transfer

- 1 SPI at each end
- Many DPIs per SPI
- Each DPI transfers a portion of the file
- Allows for fast transfers
- Many NICs per transfer

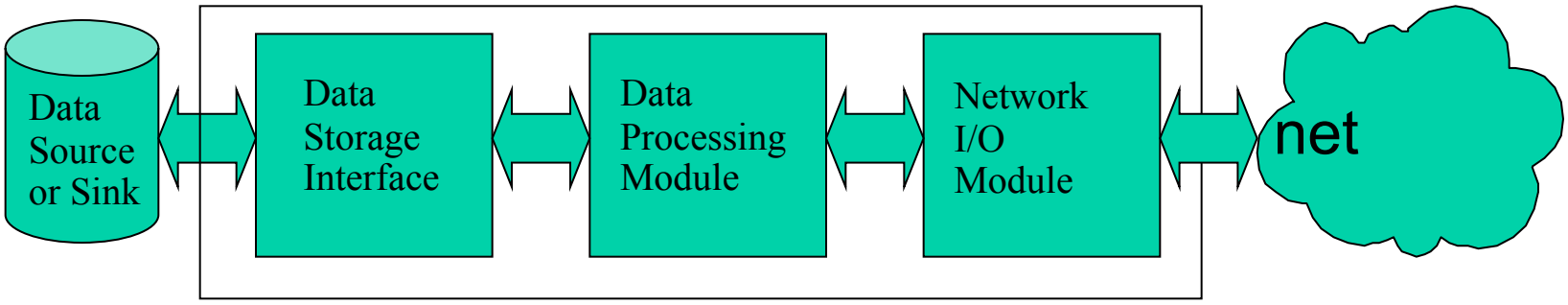
Cluster-to-cluster transfer



Modular

Globus GridFTP is based on XIO and is modular

Well-defined interfaces



Data Storage Interface (DSI)

Number of storage systems in use by the scientific and engineering community

- High Performance Storage System (HPSS)
- Distributed File System (DFS)
- Storage Resource Broker (SRB)

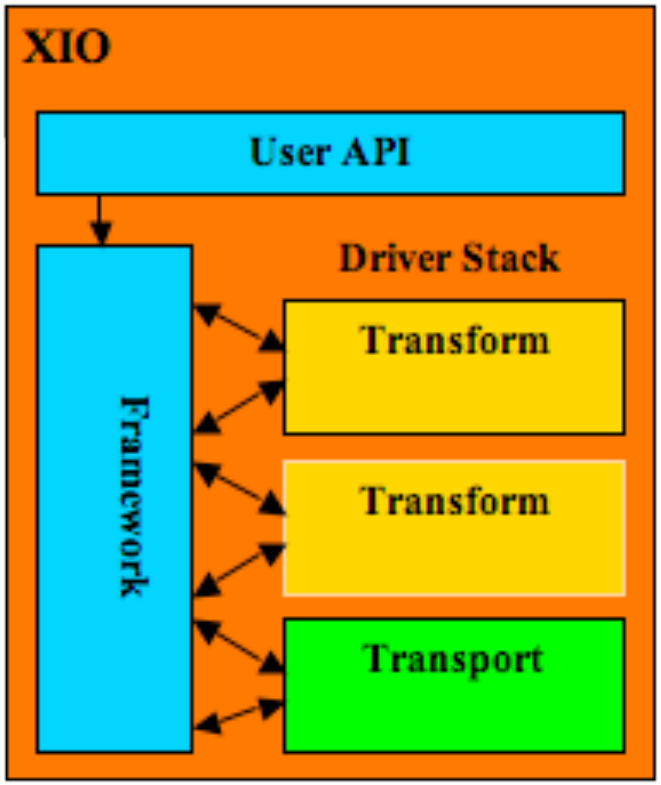
Use incompatible protocols for accessing data and require the use of their own clients

Modular abstraction to storage systems

the globus alliance
www.globus.org

Globus XIO

- Framework to compose different protocols
- Provides a unified interface open/close/read/write
- Driver interface to hook 3rd party protocol libraries



Alternative stacks

All I/O in GridFTP is done with Globus XIO

- data channel and disk

XIO allows you to set an I/O software stack

- transport and transform drivers
- ex: compression, gsi,tcp

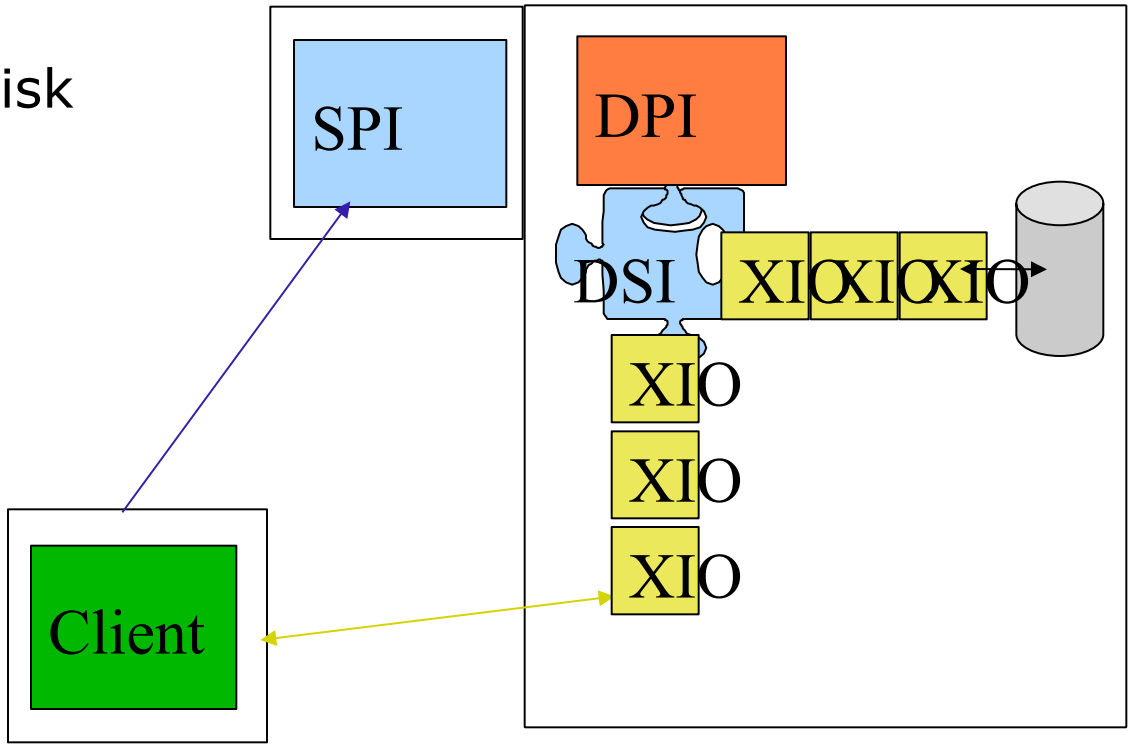
Substitute UDT for TCP

Add BW limiting, or netlogger

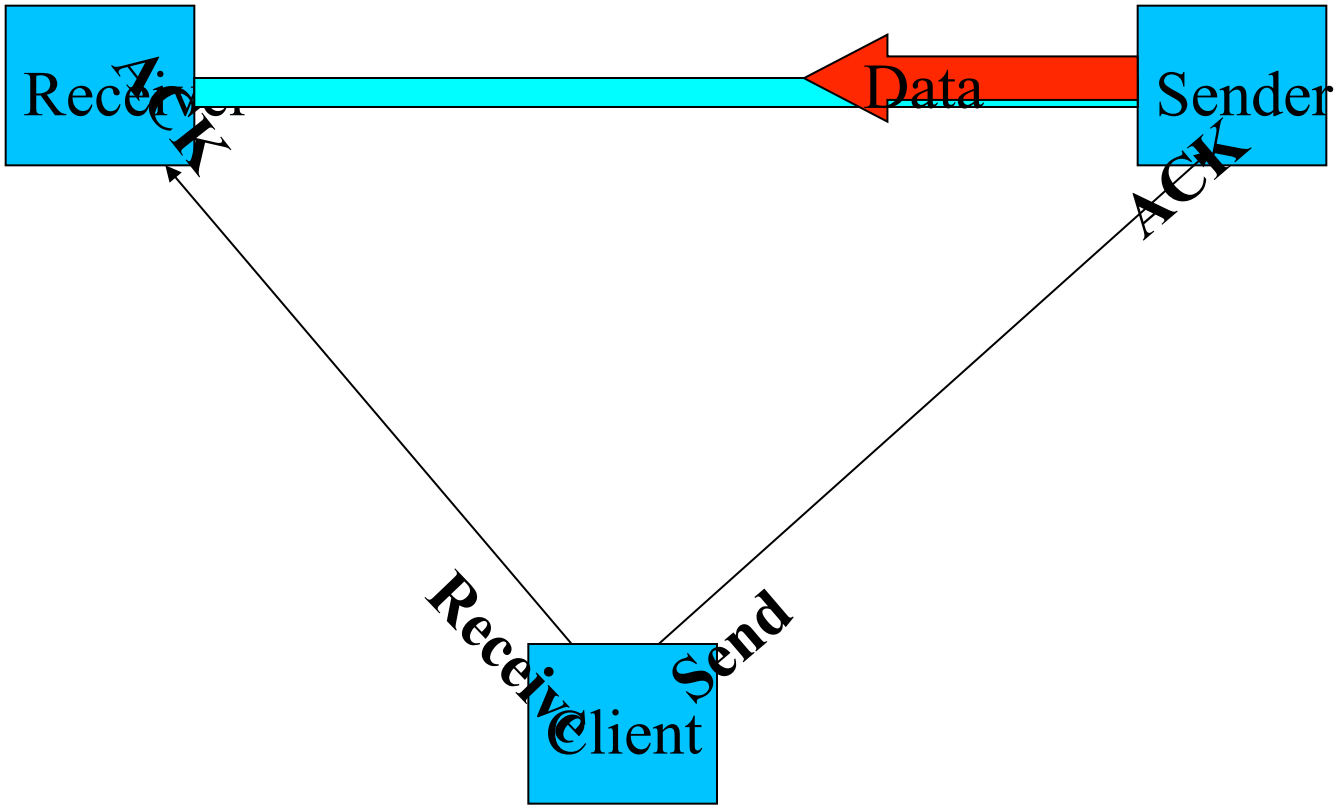
XIO Driver Stacks

All data passes through XIO driver stacks

- to network and disk
- observe data
- change data
- change protocol



Lots of Small Files (LOSF) Problem



Concurrency

Use concurrency optimization for transferring lots of small files

What is a small file?

- Depends on the network bandwidth and latency
- Files of size <= 100 MB

Transfer multiple files concurrently

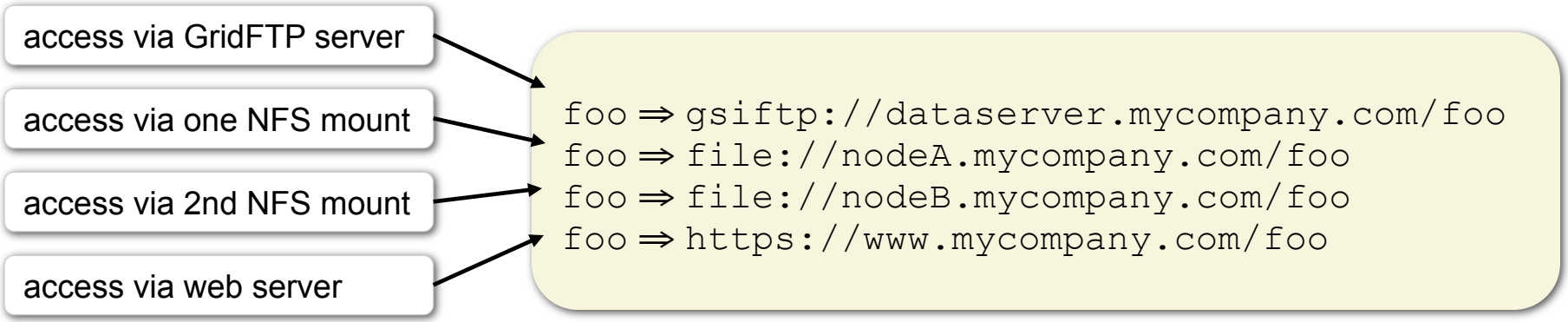
- globus-url-copy -cc

Globus Replica Location Service

- Why replicate files?
  - Fault tolerance: avoid single points of failure
  - Reduce latency: use "nearest" copy
- Logical File Name (LFN)
  - Location-independent identifier (name)
  - Example: `foo`
- Physical File Name (PFN)
  - Specific file identifier such as a URL
  - E.g.: `gsiftp://myserver.mycompany.com/foo`
- RLS maps between LFNs and PFNs
  - `foo` ⇒ `gsiftp://myserver.mycompany.com/foo`

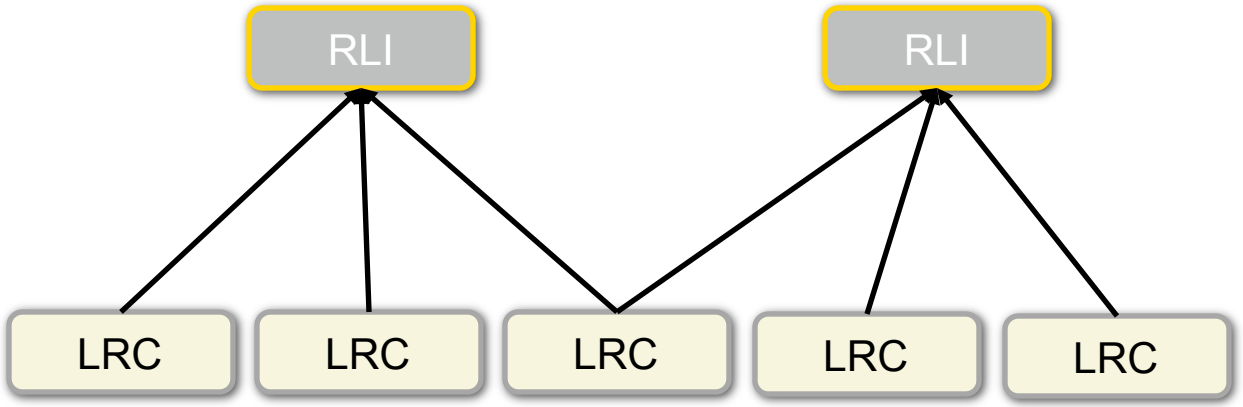
LFNs and PFNs

- LFN to PFN mappings are often many-to-one
- Multiple PFNs may indicate different access to a file



RLS services

- Local replica catalog (LRC): Catalog of LFN to PFN mappings
- Replica Location Index (RLI): Aggregate information about one or more LRCs
- Only the LFN content for LRC is aggregated
  - Each configured LRC sends list of LFNs to LRCs
  - PFNs and mappings **not** aggregated



OGSA-DAI

- Grid Interfaces to Databases
  - Data access
    - > Relational & XML Databases, semi-structured files
  - Data integration
    - > Multiple data delivery mechanisms, data translation

More Specifically, I May Want To …

- Manage who is allowed to access my service (or my experimental data or …)
- Ensure reliable & secure distribution of data from my lab to my partners
- Run 10,000 jobs on whatever computers I can get hold of

Execution Management (GRAM)

GRAM is a Globus Toolkit component

- For Grid *job management*

GRAM is a unifying remote interface to Resource Managers

- Yet preserves local site security/control

GRAM provides stateful job control

- Reliable create operation
- Asynchronous monitoring and control
- Remote credential management
- Remote file staging and file cleanup

Grid Job Management Goals

Provide a service to securely:

Create an environment for a job

Stage files to/from environment

Cause execution of job process(es)

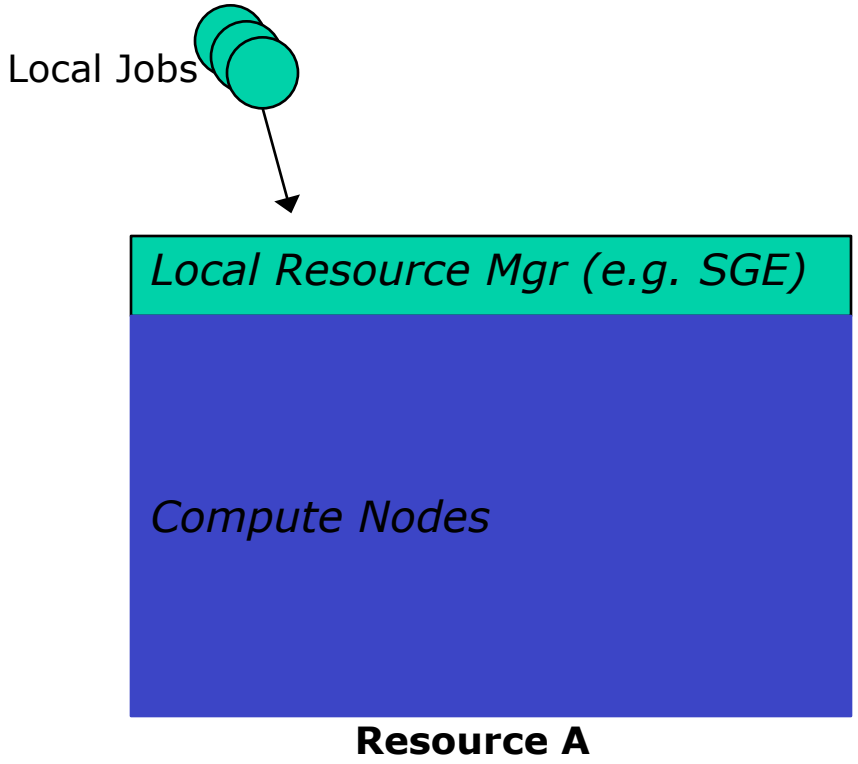
- Via various local resource managers

Monitor execution

Signal important state changes to client

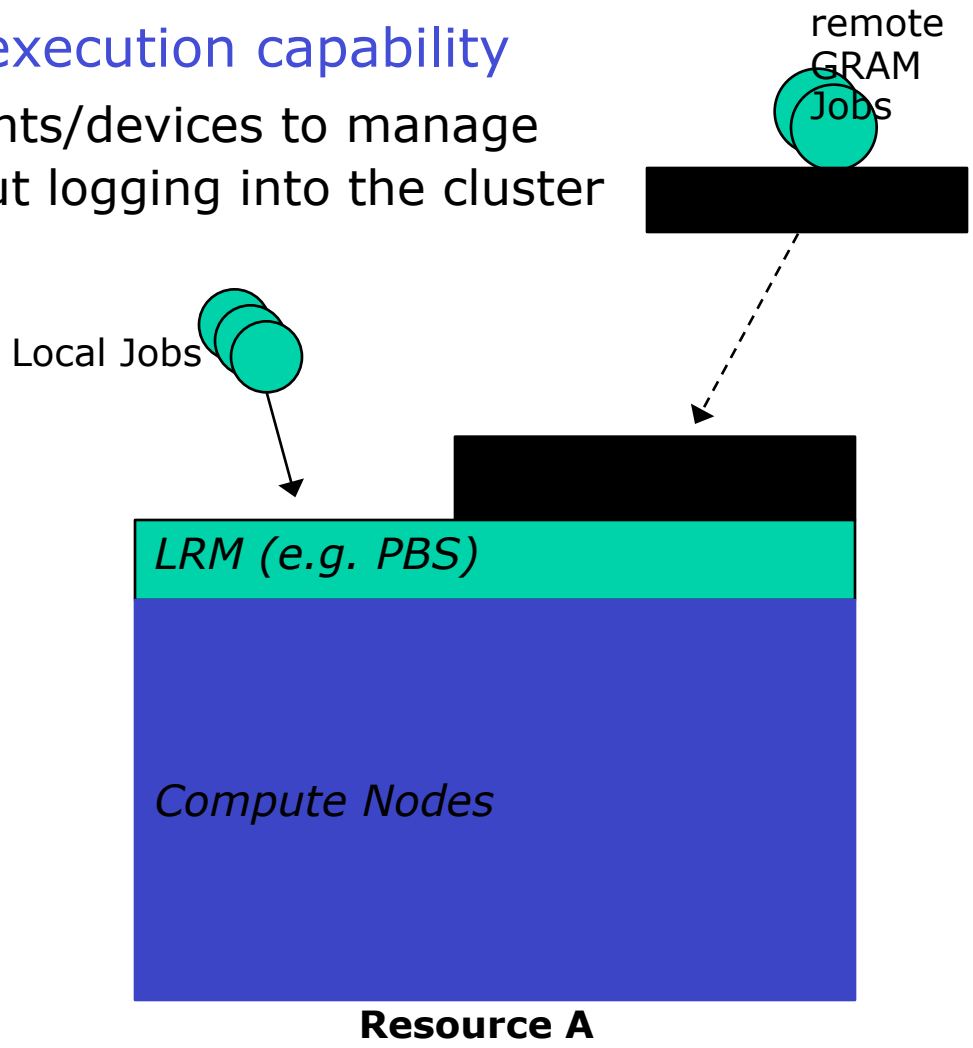
Traditional Interaction

- Satisfies many use cases
- TACC's Ranger (62976 cores!) is the Costco of HTC ;-), one stop shopping, why do we need more?



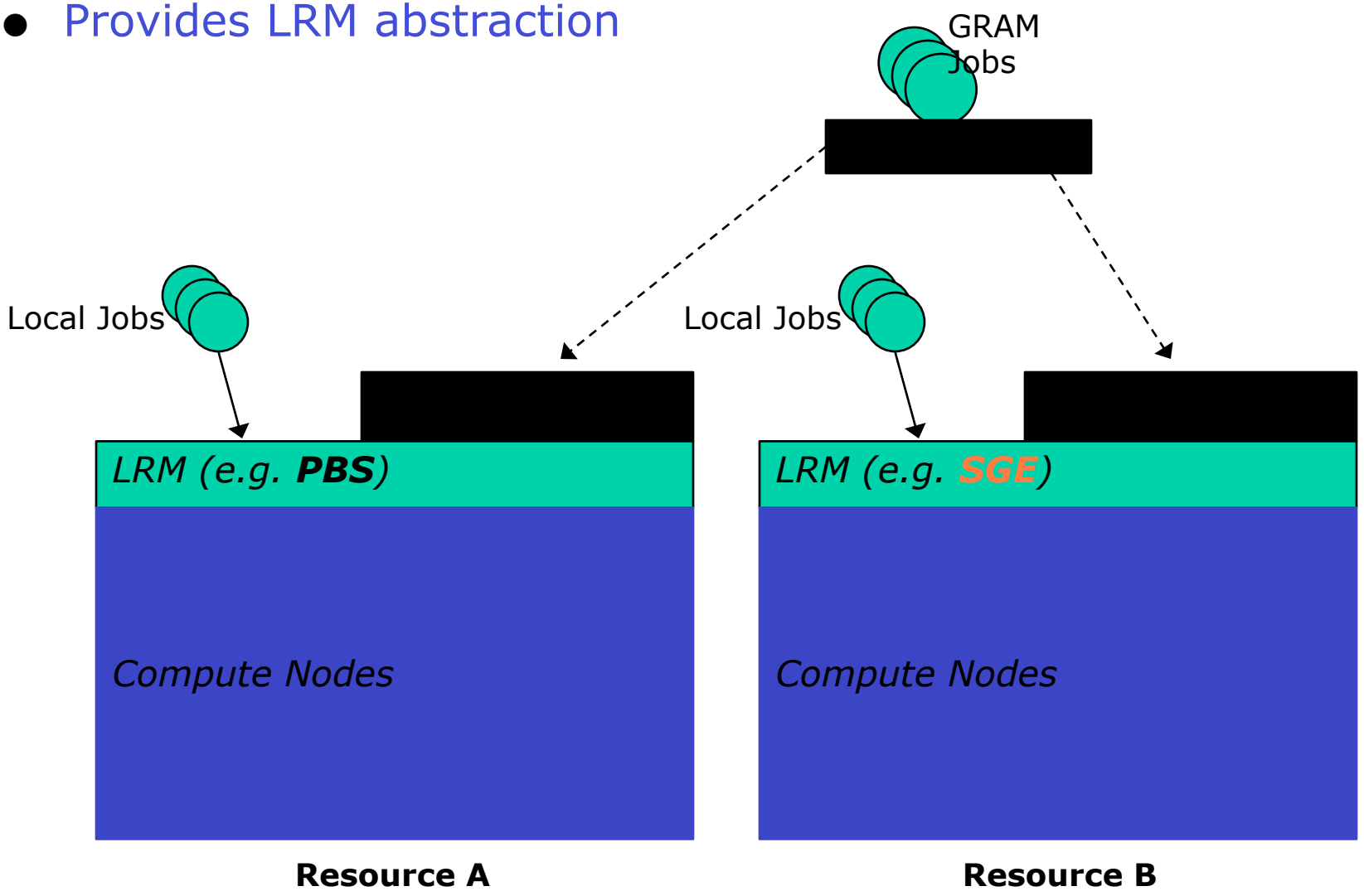
GRAM Benefit

- Add remote execution capability
  - Enable clients/devices to manage jobs without logging into the cluster



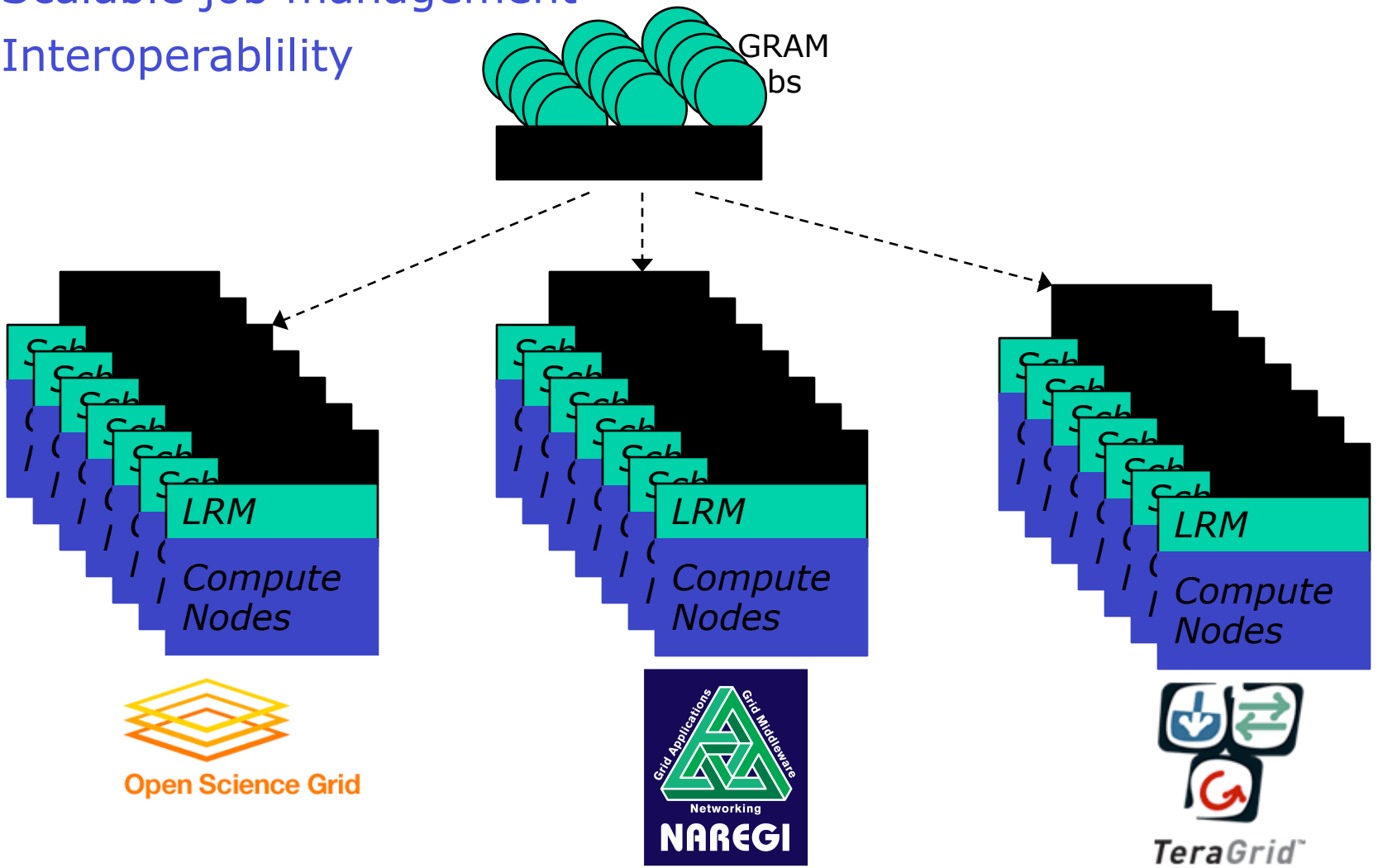
GRAM Benefit

- Provides LRM abstraction

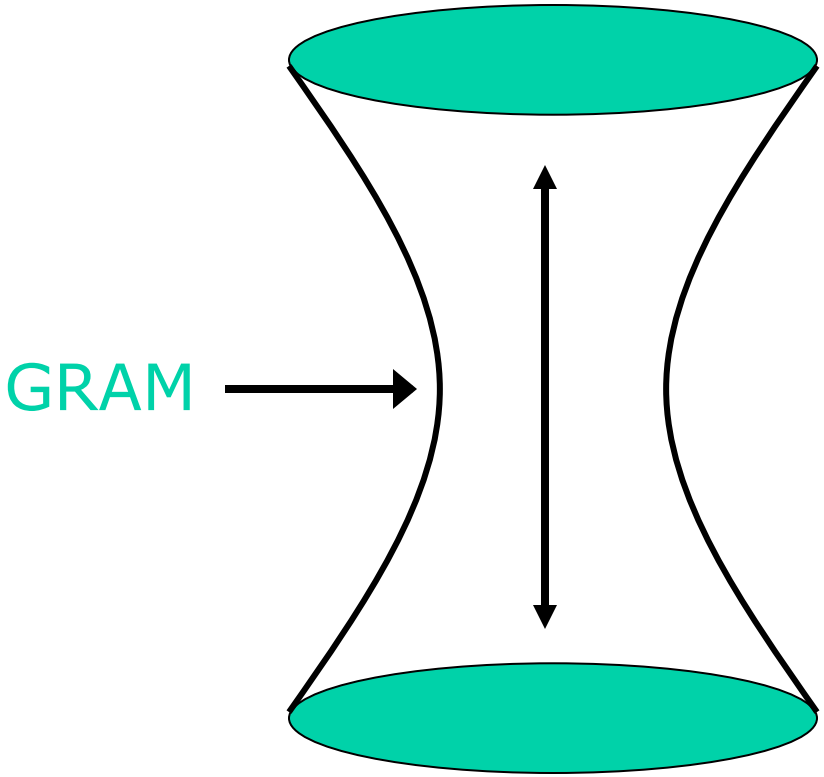


GRAM Benefit

- Scalable job management
- Interoperablility



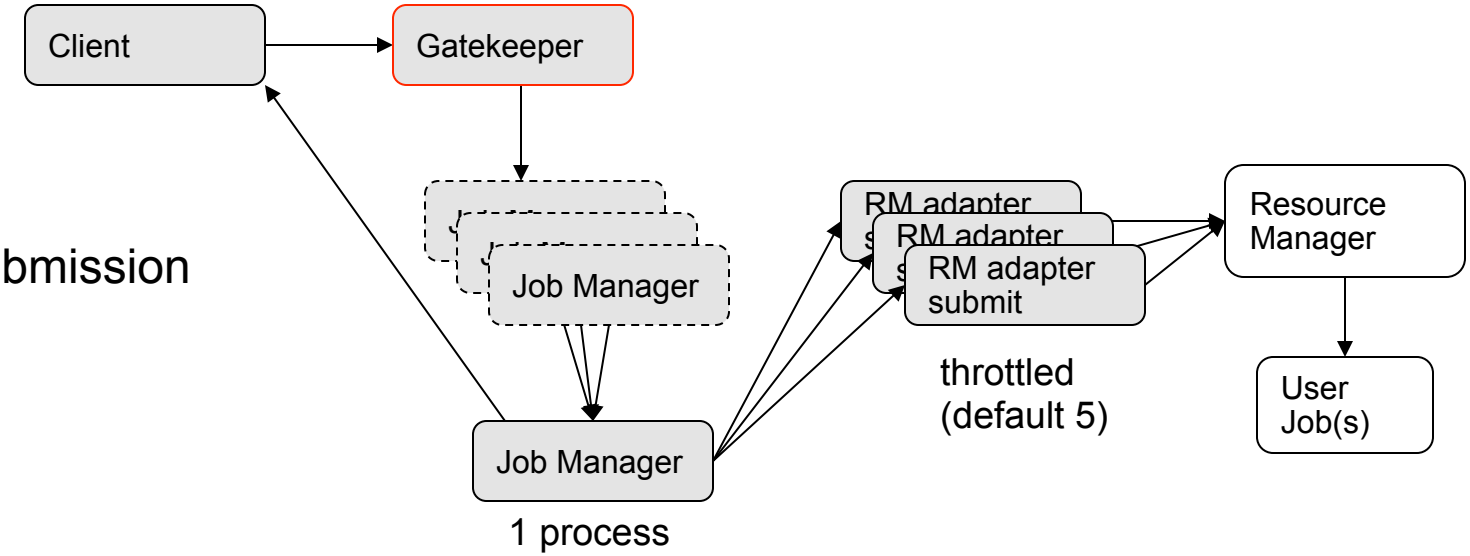
Users/Applications:
Science Gateways, Portals, CLI scripts,
App Specific Web Service, etc.



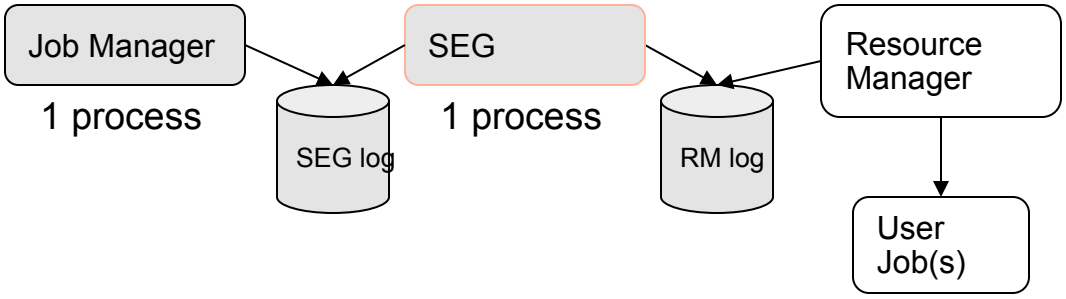
Local Resource Managers:
PBS, Condor, LSF, SGE, Fork

GRAM5 Architecture

Job Submission

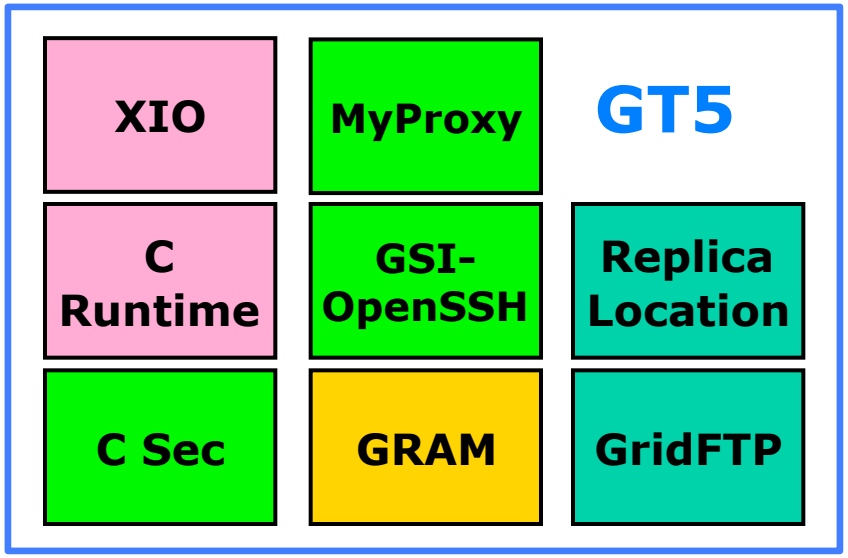


Job Monitoring



The GT5 Distribution

Globus Toolkit Version 5



Installation in a nutshell

- Quickstart guide is very useful http://www.globus.org/toolkit/docs/5.0/5.0.0/admin/quickstart/
- Verify your prereqs!
- Security – check spellings and permissions
- Globus is system software – plan accordingly

General Globus Help and Support

- Globus toolkit help lists list
  - gt-user@globus.org
  - gt-dev@globus.org
  - http://dev.globus.org/wiki/Mailing_Lists
- Each project has specific lists
- Project tracking
  - jira.globus.org

GridWay Meta-Scheduler

- Scheduler virtualization layer on top of Globus services
  - A LRM-like environment for submitting, monitoring, and controlling jobs
  - Submit jobs to the Grid, without having to worry about the details of exactly which local resource will run the job
  - A policy-driven job scheduler
  - Accounting
  - Fault detection & recovery
  - Arrays of jobs, DAGs

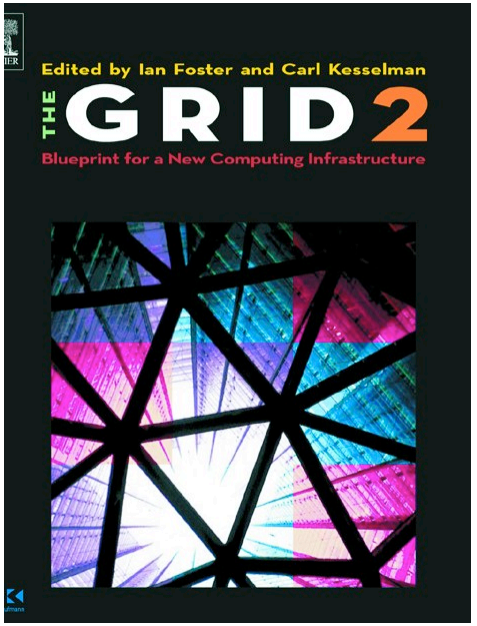
Contribute to an Existing Project

- Contribute code, documentation, design ideas, and feature requests
- Joining the mailing lists
  - *-dev, *-user, *-announce for each project
  - See the project wiki page at dev.globus.org
- Chime in at any time
- Regular contributors can become committers, with a role in defining project directions

http://dev.globus.org/wiki/How_to_contribute

the globus alliance
www.globus.org

Summary: Grids are About …



Enabling *"coordinated resource sharing & problem solving in dynamic, multi-institutional virtual organizations."*

(Source: **"The Anatomy of the Grid"**)

- Access to shared resources

→ Virtualization, allocation, management

- With predictable behaviors

→ Provisioning, quality of service

- In dynamic, heterogeneous environments

→ Standards-based interfaces and protocols

… By Providing Open Infrastructure

- Services that enable access to resources
  - Service-enable new & existing resources
  - E.g., GRAM on computer, GridFTP on storage system, custom application services
  - Uniform abstractions & mechanisms
- Tools to build applications that exploit this infrastructure
  - Registries, security, data management, …
- A rich tool & service ecosystem

More Specifically, Making it Possible to …

- Manage who is allowed to access my service (or my experimental data or …)
- Ensure reliable & secure distribution of data from my lab to my partners
- Run 10,000 jobs on whatever computers I can get hold of
- And so on …

For More Information

- Globus Alliance
  - http://www.globus.org
- Dev.globus
  - http://dev.globus.org
- Upcoming Events
  - http://dev.globus.org/wiki/Outreach
- Globus Solutions
  - http://www.globus.org/solutions/